

CERTIFIED COPY OF
PRIORITY DOCUMENT

日 本 国 特 許 庁
PATENT OFFICE
JAPANESE GOVERNMENT

JC882 U.S. PTO

09/736357



別紙添付の書類に記載されている事項は下記の出願書類に記載されて
る事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出 願 年 月 日
Date of Application:

2000年 2月21日

願 番 号
Application Number:

特願2000-043441

願 人
Applicant(s):

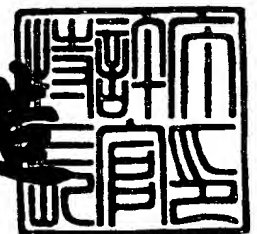
富士通株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

2000年10月20日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3086278

【書類名】 特許願

【整理番号】 9940973

【提出日】 平成12年 2月21日

【あて先】 特許庁長官 近藤 隆彦 殿

【国際特許分類】 G06F 9/312

【発明の名称】 計算機とその制御方法

【請求項の数】 4

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 三宅 英雄

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 須賀 敦浩

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 中村 泰基

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100070150

【住所又は居所】 東京都渋谷区恵比寿4丁目20番3号 恵比寿ガーデンプレイスタワー32階

【弁理士】

【氏名又は名称】 伊東 忠彦

【電話番号】 03-5424-2511

【手数料の表示】

【予納台帳番号】 002989

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9704678

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 計算機とその制御方法

【特許請求の範囲】

【請求項1】 プログラムされた命令の実行を制御する計算機の制御方法であって、

前記プログラムにおいて分岐命令より後置された命令を、前記分岐命令より先に実行する第一のステップと、

前記第一のステップにおいて、例外動作の必要性が検出された場合には、前記例外動作を留保する第二のステップと、

前記分岐命令を実行することにより選択された分岐先の命令の実行において、前記第二のステップで留保した前記例外動作が必要とされる場合には、前記例外動作を遂行する第三のステップと、

前記例外動作が終了したときには、前記プログラムに復帰して前記分岐先の命令の実行を続行する第四のステップとを有することを特徴とする計算機の制御方法。

【請求項2】 プログラムされた命令の実行を制御する計算機の制御方法であって、

前記プログラムにおいて分岐命令より後置された命令を、前記分岐命令より先に実行する第一のステップと、

前記第一のステップにおいて、例外動作を必要とする例外起因命令が検出された場合には、前記例外動作を留保する第二のステップと、

前記分岐命令を実行することにより選択された分岐先の命令の実行において、前記第二のステップで留保した前記例外動作が必要とされる場合には、前記例外動作を遂行する第三のステップと、

前記例外動作が終了したときには、前記プログラムに復帰して前記例外起因命令から順次実行する第四のステップとを有することを特徴とする計算機の制御方法。

【請求項3】 プログラムされた命令を実行する計算機であって、

前記プログラムにおいて分岐命令より後置された命令を、前記分岐命令より先

に実行するよう制御する制御手段と、

前記後置された命令の実行中に、例外動作の必要性が検出された場合には、前記例外動作を留保する例外抑止手段と、

前記分岐命令を実行することにより選択された分岐先の命令の実行において、前記例外抑止手段により留保された前記例外動作が必要とされる場合には、前記例外動作を遂行する例外動作実行手段と、

前記例外動作が終了したときには、前記プログラムに復帰して前記分岐先の命令の実行を続行する復帰手段とを備えたことを特徴とする計算機。

【請求項 4】 プログラムされた命令を実行する計算機であって、

前記プログラムにおいて分岐命令より後置された命令を、前記分岐命令より先に実行するよう制御する制御手段と、

前記後置された命令の実行中に、例外動作を必要とする例外起因命令が検出された場合には、前記例外動作を留保する例外抑止手段と、

前記分岐命令を実行することにより選択された分岐先の命令の実行において、前記例外抑止手段により留保された前記例外動作が必要とされる場合には、前記例外動作を遂行する例外動作実行手段と、

前記例外動作が終了したときには、前記プログラムに復帰して前記例外起因命令から順次実行する復帰手段とを備えたことを特徴とする計算機。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、プログラムされた命令を実行する計算機とその制御方法に関するものである。

【0 0 0 2】

【従来の技術】

図 1 は、汎用レジスタと浮動小数点レジスタとを有する従来の計算機の構成を示す図である。図 1 に示されるように、この計算機はメモリ 1 と、メモリ 1 に接続された命令読み出し部 3 と、メモリ 1 及び命令読み出し部 3 に接続された命令実行部 6 と、命令実行部 6 に接続されたレジスタ制御部 8 と、命令読出し部 3 と

命令実行部 6 及びレジスタ制御部 8 に接続された割り込み制御部 9 とを備える。

ここで、命令読み出し部 3 は命令読み出し制御部 1 1 と、プログラムカウンタ (PC) 1 3 と、命令語レジスタ (IR) 1 5 とを含む。そして、命令読み出し制御部 1 1 はメモリ 1 に接続され、プログラムカウンタ (PC) 1 3 は命令読み出し制御部 1 1 に接続される。さらに、命令語レジスタ (IR) 1 5 は命令読み出し制御部 1 1 に接続される。

【 0 0 0 3 】

また、命令実行部 6 は命令デコード部 1 7 と、ロード命令実行部 1 9 と、ストア命令実行部 2 1 と、演算命令実行部 2 2 と、命令実行回路 2 3 と、浮動小数点 (F) ロード命令実行部 2 5 と、浮動小数点 (F) ストア命令実行部 2 7 と、浮動小数点 (F) 演算命令実行部 2 9 とを含む。

【 0 0 0 4 】

そして、命令デコード部 1 7 は命令語レジスタ 1 5 に接続され、ロード命令実行部 1 9 はメモリ 1 及び命令デコード部 1 7 に接続される。また、ストア命令実行部 2 1 は命令デコード部 1 7 及び後述する汎用レジスタ 3 7 に接続され、演算命令実行部 2 2 は命令デコード部 1 7 と汎用レジスタ 3 7 及び後述する条件レジスタ 3 0 に接続され、命令実行回路 2 3 は命令デコード部 1 7 と汎用レジスタ 3 7 及び後述するレジスタ 3 1, 3 3, 3 5 に接続される。

【 0 0 0 5 】

また、浮動小数点ロード命令実行部 2 5 はメモリ 1 及び命令デコード部 1 7 に接続され、浮動小数点ストア命令実行部 2 7 及び浮動小数点演算命令実行部 2 9 は命令デコード部 1 7 及び後述する浮動小数点レジスタ 3 9 に接続される。

【 0 0 0 6 】

一方、レジスタ制御部 8 は条件レジスタ 3 0 と、レジスタ (EPCR) 3 1 と、レジスタ (EPSR) 3 3 と、レジスタ (PSR) 3 5 と、汎用レジスタ 3 7 と、浮動小数点レジスタ 3 9 とを含む。ここで、条件レジスタ 3 0 は演算命令実行部 2 2 と命令実行回路 2 3 及び浮動小数点演算命令実行部 2 9 に接続され、レジスタ (EPCR) 3 1 とレジスタ (EPSR) 3 3 及びレジスタ (PSR) 3 5 は、さらに割込制御部 4 0 に接続される。また、汎用レジスタ 3 7 はロード命

令実行部 1 9 とストア命令実行部 2 1 及び命令実行回路 2 3 に接続される。また、浮動小数点レジスタ 3 9 は浮動小数点ロード命令実行部 2 5 と、浮動小数点ストア命令実行部 2 7 及び浮動小数点演算命令実行部 2 9 に接続される。

【 0 0 0 7 】

また、割り込み制御部 9 は割込制御回路 4 0 を含む。そして、この割込制御回路 4 0 は命令読出制御部 1 1 とプログラムカウンタ 1 3、ロード命令実行部 1 9、ストア命令実行部 2 1、演算命令実行部 2 2、命令実行回路 2 3、浮動小数点ロード命令実行部 2 5、浮動小数点ストア命令実行部 2 7 及び浮動小数点演算命令実行部 2 9 などに接続される。

【 0 0 0 8 】

上記のような構成を有する計算機において、命令読み出し部 3 はプログラムカウンタ 1 3 が示す命令語をメモリ 1 から読み出し、命令語レジスタ (I R) 1 5 を介して命令実行部 6 へ供給する。また命令読出制御部 1 1 は、命令実行部 6 あるいは割込処理を行う割込制御回路 4 0 より分岐先アドレスが供給された場合には、それをプログラムカウンタ 1 3 へ書き込む。なお命令読出制御部 1 1 は、それ以外の場合には次の命令語を命令実行部 6 へ供給するため、読み出すべき命令語のアドレスを示すプログラムカウンタ 1 3 をインクリメントする。また、命令語を読み出す際に割り込みを検出した場合には、割込制御回路 4 0 へ割込信号を供給する。

【 0 0 0 9 】

また、命令デコード部 1 7 は、命令語レジスタ 1 5 から供給された命令をデコードし、ロード命令の場合にはロード命令実行部 1 9 へ、ストア命令の場合にはストア命令実行部 2 1 へ、演算命令の場合には演算命令実行部 2 2 へ、浮動小数点ロード命令の場合には浮動小数点ロード命令実行部 2 5 へ、浮動小数点ストア命令の場合には浮動小数点ストア命令実行部 2 7 へ、浮動小数点演算命令の場合には浮動小数点演算命令実行部 2 9 へ、割込復帰命令やその他の命令の場合には命令実行回路 2 3 へそれぞれ供給する。

【 0 0 1 0 】

また、ロード命令実行部 1 9 はロード命令が供給された場合には、図 2 に示さ

れるように、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応するメモリ 1 の領域からデータを読み出し、その結果を汎用レジスタ 3 7 へ書き込む。ここでロード命令は、図 3 に示されるように、命令コード OP-CODE とレジスタを指定するコード GR1, GR2, GRD とを含む。そして、コード GR1 が示すレジスタの値とコード GR2 が示すレジスタの値を加算した結果がロード対象とするデータのアドレスを意味し、コード GRD は読み出した結果を保持するレジスタの番号を意味する。なお、ロード命令を実行する際に割込みを検出した場合には、割込信号を割込制御回路 4 0 へ供給する。

【 0 0 1 1 】

同様に、ストア命令実行部 2 1 はストア命令が供給された場合には、図 4 に示されるように、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応する汎用レジスタ 3 7 の領域からデータを読み出し、その結果を実効アドレスに対応するメモリ 1 の領域へ書き込む。ここでストア命令は、図 5 に示されるように、命令コード OP-CODE とレジスタを指定するコード GR1, GR2, GR S とを含む。そして、コード GR1 が示すレジスタの値とコード GR2 が示すレジスタの値を加算した結果がストア対象とするデータのアドレスを意味し、コード GR S は書き込む値を保持するレジスタの番号を意味する。なお、ストア命令を実行する際に割込みを検出した場合には、割込信号を割込制御回路 4 0 へ供給する。

【 0 0 1 2 】

また、演算命令実行部 2 2 は演算命令が供給された場合には、図 6 に示されるように、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応する汎用レジスタ 3 7 の領域からデータを読み出し、そのデータに基づき演算し、該演算結果を汎用レジスタ 3 7 へ書き込む。なお演算命令は、図 3 に示された上記ロード命令と同じ形式を有する。さらに演算命令実行部 2 2 は、比較命令が供給された場合には、汎用レジスタ 3 7 から読み出された二つの値を比較し、該二つの値が等しい場合には真であることを示すデータを、該二つの値が等しくない場合には偽であることを示すデータを条件レジスタ 3 0 へ書き込む。

【 0 0 1 3 】

また、浮動小数点ロード命令実行部 2 5 は、浮動小数点ロード命令が供給され

た場合には、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応するメモリ 1 の領域からデータを読み出し、その結果を浮動小数点レジスタ 3 9 へ書き込む。なお、浮動小数点ロード命令を実行する際に割込みを検出した場合には、割込信号を割込制御回路 4 0 へ供給する。

【 0 0 1 4 】

また、浮動小数点ストア命令実行部 2 7 は、浮動小数点ストア命令が供給された場合には、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応する浮動小数点レジスタ 3 9 の領域からデータを読み出し、その結果を実効アドレスに対応するメモリ 1 の領域へ書き込む。なお、浮動小数点ストア命令を実行する際に割込みを検出した場合には、割込信号を割込制御回路 4 0 へ供給する。

【 0 0 1 5 】

また、浮動小数点演算命令実行部 2 9 は、浮動小数点演算命令が供給された場合には浮動小数点レジスタ 3 9 から読み出した値に基づき演算を実行し、その結果を浮動小数点レジスタ 3 9 へ書き込む。そしてさらに、浮動小数点演算命令実行部 2 9 は、浮動小数点比較命令が供給された場合には浮動小数点レジスタ 3 9 から読み出した二つの値を比較し、その結果として、該二つの値が等しいか否かにより真又は偽を示す値を条件レジスタ 3 0 へ書き込む。

【 0 0 1 6 】

また、命令実行回路 2 3 は、命令デコード部 1 7 から分岐命令が供給された場合には、分岐成立時に分岐先アドレスをプログラムカウンタ 1 3 へ供給する。さらに、命令デコード部 1 7 から条件分岐命令が供給された場合には、条件レジスタ 3 0 に格納された値が真であることを示すものである時に分岐先アドレスをプログラムカウンタ 1 3 へ供給する。同様に割込み復帰命令が供給された場合には、割込み発生前の動作状態を示すデータを P S R 3 5 へ書き込む。そしてこのとき、E P C R 3 1 から復帰先の命令のアドレスを読み出し、それを分岐先アドレスとしてプログラムカウンタ 1 3 へ供給する。なお、上記命令を実行する際に割込みを検出した場合には、割込信号を割込制御回路 4 0 へ供給する。

【 0 0 1 7 】

また、条件レジスタ 3 0 は比較命令に応じて真あるいは偽を示すデータ（値）を保持する。なお、該値は条件分岐命令に応じて参照される。また、E P C R 3 1 は割り込みからの復帰先に該当する命令のアドレスを保持し、そのアドレスは割り込み発生時に設定される。また、P S R 3 5 は動作状態（state）を示すデータを保持し、E P S R 3 3 は割り込み発生時に設定された割り込み発生前の動作状態を示すデータを保持する。

【 0 0 1 8 】

また、割込制御回路 4 0 は命令読み出し部 3 あるいは命令実行部 6 から供給された割込信号に基づいて、割り込みからの復帰先に該当する命令のアドレスを E P C R 3 1 へ、割り込み発生前の動作状態を示すデータを E P S R 3 3 へ、発生した割り込みに対応する動作状態を示す P S R 3 5 へ書き込む。また、発生した割り込みに対応する分岐先アドレスを命令読み出し部 3 へ供給する。

【 0 0 1 9 】

以下に、上記計算機の動作を要約すると、初期動作状態を通常の動作としたとき次のようになる。通常の動作においては、命令読み出し部 3 は、プログラムカウンタ 1 3 が示す命令語を読み出し、命令実行部 6 へ供給し、命令実行部 6 は供給された命令を実行する。

【 0 0 2 0 】

一方、割り込みが発生した時には、割込制御回路 4 0 は命令読み出し部 3 あるいは命令実行部 6 から供給された割込信号に基づいて、割り込みからの復帰先に該当する命令のアドレスを E P C R 3 1 へ、割り込み発生前の動作状態を示すデータを E P S R 3 3 へ、発生した割り込みに対応する動作状態を示す P S R 3 5 へ書き込む。また、発生した割り込みに対応する分岐先アドレスを命令読み出し部 3 へ供給する。また、命令読み出し部 3 は割込制御部 9 から供給された分岐先アドレスに応じて命令語を読み出し、命令実行部 6 へ供給する。そして、以降においては上記の通常の場合と同様に動作する。

【 0 0 2 1 】

また、割り込みから復帰する時は、命令実行部 6 は割り込み復帰命令を実行することにより、E P S R 3 3 の値を P S R 3 5 へ書き込む。また、E P C R 3 1 に

格納されたデータを読み出し、その結果を分岐先アドレスとして命令読み出し部 3 へ供給する。このとき命令読み出し部 3 は、命令実行部 6 から供給された分岐先アドレスに基づいて命令語を読み出し、命令実行部 6 へ供給する。そして、以降においては上記通常の動作がなされる。

【 0 0 2 2 】

以上のような従来の計算機では、プログラムを構成する命令語の並列性を活用し、計算機の命令レベルでの処理能力を向上させる手法として、スーパースカラや投機的実行 (speculative execution) といった、同時に複数の命令を実行する制御方式が採られている。ここで、一般にこのような計算機では、複数の命令実行部を有し、プログラムに含まれる命令を順次実行するが、1 サイクルにおいて複数の命令がメモリから読み出され、また該命令相互の依存関係が考慮されつつ 1 サイクル中で複数の命令が発行されうる。

【 0 0 2 3 】

また、この時命令の実行制御において、アウト・オブ・オーダー完了 (out-of-order completion) を採ることで、計算機の命令レベルでの処理能力をより向上させることができる。ここで、「アウト・オブ・オーダー完了」とは、プログラム上における命令の順序すなわち命令の発行順序と、命令の実行順序すなわち命令の完了順序とが異なることを意味する。そして、このような実行制御を行うことにより、命令実行部の有効稼働率を高め、プログラムの全体的な実行時間の短縮が図られている。なお、アウト・オブ・オーダー完了において、プログラムが生成された時の命令の順序を保証するためには、データ依存関係や制御依存関係といった命令の依存関係が考慮されるが、これらの依存関係は命令語に記されている情報から抽出される。

【 0 0 2 4 】

さらに、上記のような計算機の性能をより向上させるべく、コンパイラ他によってプログラムが生成された時に命令の順序を変更するといった手法が知られており、このような手法を「命令移動」という。そして、この「命令移動」は命令実行部の有効稼働率を高める点や、早期に命令の実行を開始することによりレイテンシを隠蔽できる点で有効な手法とされる。すなわち、例えばロード命令を実

行する場合には、メモリへのロード操作を早期に開始することで、潜在的なキャッシュミスを含むメモリアクセスの遅延を隠蔽できる可能性がある。

【 0 0 2 5 】

ここで、上記のような命令移動において、特にコンパイラにおける基本ブロックの境界を越えた命令移動を行うものは、広域命令移動と呼ばれており、さらに広域命令移動において条件分岐を越えて命令移動を行うものは、投機的命令移動と呼ばれている。しかし、このように投機的に移動した命令において例外が発生した場合、例外処理を必要とするため大幅な性能の低下を招いたり、本来のプログラムの順序ではあり得なかったプログラム実行における不正な停止を招いたりする。

【 0 0 2 6 】

すなわち例えば、必要とされるデータがメモリ上に無い状態であるページフォルト (page fault) を起こす可能性がある命令を投機的に移動した場合には、移動先においてページフォルトを伴う例外処理プログラムが実行されることとなり、本来の順序では生じ得なかったページフォルトによって性能が低下する。また、例えば除算命令を投機的に移動した場合には、移動先においてゼロで除算する処理 (ゼロ除算) が発生することも生じるが、このような場合にはプログラムの実行が停止することとなる。なお上記のように、投機的に移動した命令の実行において発生した例外は投機的例外と呼ばれる。

【 0 0 2 7 】

そして、上記問題を解決する方法として、例外抑止 (non-excepting) 命令を用いて投機的例外の発生を延期する方法については知られている。すなわち、投機的例外処理を延期する例外抑止命令方式について、文献 "A VLIW architecture for a trace scheduling compiler", Proceedings of the 2nd International Conference on Architectural Support for Programming Languages and Operating Systems, pp. 180-192, 1987 (B. P. Colwel, B. P. Nix, J. J. O'Donnell, D. B. Papworth and P. K. Rodman 共著) が発行されており、例外抑止命令方式を拡張し、投機的例外からの実行の再開を実現したスケジューリングについて、文献 "Sentinel Scheduling for VLIW and Superscalar Processor", Proc

eedings of the Fifth International Conference on Architectural Support for Programming Languages and Operating Systems, pp. 238-247, 1992 (S. A. Mahlke, W. Y. Chen, W. W. Hwu, B. R. Rau and M. S. Schlansker 共著) が発行されている。

【 0 0 2 8 】

一方、上記例外にはページフォルトのように、例外要因を解消することにより本来の処理が続行可能なものと、ゼロ除算のように本来の処理を続行することが不可能なものがある。そして、上記の例外抑止命令によるデータの処理方式においては、いずれの例外も投機的例外として検出され、所定期間延期された後に実行される。

【 0 0 2 9 】

しかしながら、上記において続行可能な例外処理及びそれに続く本来の処理は、割り込み処理プログラムを実行することによりなされていたため、全体としてプログラムが長大化する結果、計算機に必要な記憶容量が増大すると共に、処理速度が低下するという問題があった。

【 0 0 3 0 】

【発明が解決しようとする課題】

本発明は、上述の問題点を解消するためになされたもので、動作の高速化により処理能力が高められた計算機とその制御方法を提供することを目的とする。

【 0 0 3 1 】

【課題を解決するための手段】

上記の目的は、プログラムされた命令の実行を制御する計算機の制御方法であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行する第一のステップと、第一のステップにおいて、例外動作の必要性が検出された場合には、例外動作を留保する第二のステップと、分岐命令を実行することにより選択された分岐先の命令の実行において、第二のステップで留保した例外動作が必要とされる場合には、例外動作を遂行する第三のステップと、例外動作が終了したときには、プログラムに復帰して例外動作の遂行が必要とされた命令の次の命令から順次実行する第四のステップとを有することを特徴とする計算機

の制御方法を提供することにより達成される。このような手段によれば、例外動作が終了したときには、プログラムに復帰して例外動作の遂行が必要とされた命令の次の命令から順次実行することにより、動作の高速化を図ることができる。

【 0 0 3 2 】

また本発明の目的は、プログラムされた命令の実行を制御する計算機の制御方法であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行する第一のステップと、第一のステップにおいて、例外動作を必要とする例外起因命令が検出された場合には、例外動作を留保する第二のステップと、分岐命令を実行することにより選択された分岐先の命令の実行において、第二のステップで留保した例外動作が必要とされる場合には、例外動作を遂行する第三のステップと、例外動作が終了したときには、プログラムに復帰して例外起因命令から順次実行する第四のステップとを有することを特徴とする計算機の制御方法を提供することにより達成される。このような手段によれば、例外動作が終了したときには、プログラムに復帰して例外起因命令から順次実行するため、動作の高速化を図ることができる。

【 0 0 3 3 】

また本発明の目的は、プログラムされた命令を実行する計算機であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行するよう制御する制御手段と、後置された命令の実行中に、例外動作の必要性が検出された場合には、例外動作を留保する例外抑止手段と、分岐命令を実行することにより選択された分岐先の命令の実行において、例外抑止手段により留保された例外動作が必要とされる場合には、例外動作を遂行する例外動作実行手段と、例外動作が終了したときには、プログラムに復帰して例外動作の遂行が必要とされた命令の次の命令から順次実行する復帰手段とを備えたことを特徴とする計算機を提供することにより達成される。

【 0 0 3 4 】

また本発明の目的は、プログラムされた命令を実行する計算機であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行するよう制御する制御手段と、後置された命令の実行中に、例外動作を必要とする例外起

因命令が検出された場合には、例外動作を留保する例外抑止手段と、分岐命令を実行することにより選択された分岐先の命令の実行において、例外抑止手段により留保された例外動作が必要とされる場合には、例外動作を遂行する例外動作実行手段と、例外動作が終了したときには、プログラムに復帰して例外起因命令から順次実行する復帰手段とを備えたことを特徴とする計算機を提供することにより達成される。

【 0 0 3 5 】

【発明の実施の形態】

以下において、本発明の実施の形態を図面を参照しつつ詳しく説明する。なお、図中同一符号は、同一又は相当部分を示す。

〔実施の形態 1〕

図 7 は、本発明の実施の形態 1 に係る計算機の構成を示す図である。図 7 に示されるように、本実施の形態 1 に係る計算機は、図 1 に示された従来の計算機に対して履歴制御部 1 7 3 をさらに備え、命令実行部 1 7 0 には例外抑止ロード命令実行部 2 4、コミット命令実行部 2 6、例外抑止履歴読み出し命令実行部 2 8、例外抑止履歴書き込み命令実行部 2 0 を含み、レジスタ制御部 1 7 1 には例外抑止フラグ 3 8 を含み、割り込み制御部 1 0 にはコミット例外割り込み制御部 4 4 を含む点で相違する。

【 0 0 3 6 】

ここで、履歴制御部 1 7 3 は例外抑止履歴書き込み命令実行部 2 0 と例外抑止ロード命令実行部 2 4、コミット命令実行部 2 6、例外抑止履歴読み出し命令実行部 2 8、及びコミット例外割り込み制御部 4 4 に接続される。また、例外抑止履歴書き込み命令実行部 2 0 とコミット命令実行部 2 6、及び例外抑止履歴読み出し命令実行部 2 8 は、さらに命令デコード部 1 7 と汎用レジスタ 3 7 に接続される。そして、例外抑止ロード命令実行部 2 4 はさらにメモリ 1 に接続される。

【 0 0 3 7 】

また、例外抑止フラグ 3 8 は汎用レジスタ 3 7 に並設され、コミット例外割り込み制御部 4 4 はさらにプログラムカウンタ（PC）1 3 とレジスタ 3 1，3 3，3 5 に接続される。

【 0 0 3 8 】

上記のような構成を有する本実施の形態に係る計算機において、例外抑止ロード命令実行部 2 4 は、例外抑止ロード命令が供給された場合には、まず最初に汎用レジスタ 3 7 より読み出した値から実効アドレスを求め、該実効アドレスに対応するメモリ 1 内の領域からデータを読み出す。そして、図 9 に示されるように、ステップ S 1 において読み出したデータの中に例外要因が存在するか否か確認する。その結果、ステップ S 2 において例外要因が存在しないと判断された場合には、ステップ S 3 へ進みロード対象アドレス域から読み出した上記データを、汎用レジスタ 3 7 へ書き込む。

【 0 0 3 9 】

一方、ステップ S 2 において例外要因が存在すると判断された場合には、ステップ S 1 0 へ進み、履歴制御部 1 7 3 へ登録指示信号が供給されることにより、検出された例外情報が履歴制御部 1 7 3 に記憶される。このとき、ステップ S 1 1 において、上記例外情報が記憶されるレジスタに対応する例外抑止フラグ 3 8 に 1 の値が設定され有効とされる。なお、例外抑止ロード命令の命令形式は、図 3 に示されたロード命令の命令形式と同様なものである。

【 0 0 4 0 】

次にコミット命令実行部 2 6 は、コミット命令が供給された場合には、図 1 0 のステップ S 1 に示されるように、該命令により指定された汎用レジスタ 3 7 内のレジスタに対応する例外抑止フラグ 3 8 が有効か否か確認する。そして、ステップ S 2 において、該フラグに 1 が設定されており有効であると判断された場合には、ステップ S 3 へ進み該例外抑止フラグ 3 8 を 0 として無効にする。そして、ステップ S 4 において履歴制御部 1 7 3 へ確認指示信号が供給される。これにより、コミット例外が発生したことが割り込み制御部 1 0 へ伝達される。なお、ステップ S 2 において例外抑止フラグ 3 8 が無効であると判断された場合には、コミット命令による動作を終了する。なおコミット命令の命令形式は、図 1 1 に示されるように、命令コード OP-CODE とレジスタを指定するコード GR とからなる。

【 0 0 4 1 】

次に、例外抑止履歴読み出し命令実行部 2 8 は、例外抑止履歴読み出し命令が供給された場合、履歴制御部 1 7 3 へ読み出し指示信号を供給することにより、履歴制御部 1 7 3 に記憶されている上記例外情報を読み出し、汎用レジスタ 3 7 へ書き込む。また、同様に例外抑止履歴書き込み命令実行部 2 0 は、例外抑止履歴書き込み命令が供給された場合、汎用レジスタ 3 7 から読み出したデータと書き込み指示信号を履歴制御部 1 7 3 へ供給して、該読み出したデータを履歴制御部 1 7 3 に書き込む。

【 0 0 4 2 】

またコミット例外割り込み制御部 4 4 は、履歴制御部 1 7 3 から供給された割り込みを通知するコミット信号 CM に応じて、割り込みからの復帰先の命令アドレスをレジスタ (EPCR) 3 1 へ書き込み、割り込み発生前の動作状態を示すデータをレジスタ (EPSR) 3 3 へ書き込み、発生した割り込みに対応する動作状態をレジスタ (PSR) 3 5 へ書き込む。また、発生した割り込みに対応する分岐先アドレスをプログラムカウンタ (PC) 1 3 へ供給する。

【 0 0 4 3 】

次に履歴制御部 1 7 3 は、コミット命令実行部 2 6 から供給された確認指示信号に応じて、指定されたレジスタ番号に例外情報が保持されているか否か確認する。そして、指定されたレジスタ番号に例外情報が保持されている場合には、コミット例外を検出したことを示すコミット信号 CM をコミット例外割り込み制御部 4 4 へ供給する。

【 0 0 4 4 】

また、例外抑止履歴読み出し命令実行部 2 8 から供給された読み出し指示信号に応じて、記憶されている例外情報を読み出して例外抑止履歴読み出し命令実行部 2 8 へ供給する。さらに、例外抑止履歴書き込み命令実行部 2 0 から供給された書き込み指示信号に応じて、供給されたデータを記憶する。

【 0 0 4 5 】

図 8 は、図 7 に示された履歴制御部 1 7 3 の構成を示す図である。図 8 に示されるように、履歴制御部 1 7 3 は、アドレスレジスタ 5 7 と、データタイプレジスタ 5 9 と、レジスタ番号レジスタ 6 1 と、例外要因 (EC) レジスタ 1 7 5 と

、レジスタタイプレジスタ105と、デコード回路65と、比較器67～69と、ECフィールド (EC) 177～179と、Vフィールド151～153と、レジスタタイプフィールド (RT) 107～109と、アドレスフィールド (ADDR) 71, 75, 79と、データタイプフィールド (DT) 72, 76, 80と、レジスタ番号フィールド (REG #) 73, 77, 81と、コミット判定部180と、コミットエントリ検出部181と、無効エントリ検出部87と、例外抑止履歴登録制御部183と、コミット命令実行部185と、例外抑止履歴読み出し命令実行部187と、例外抑止履歴書き込み命令実行部189とを含む。

【0046】

ここでアドレスレジスタ57と、データタイプレジスタ59と、レジスタ番号レジスタ61と、レジスタタイプレジスタ105と、ECレジスタ175及びデコード回路65は命令実行部170に接続される。そして、アドレスレジスタ57は、例外抑止ロード命令が実行される際の実効アドレスを保持する。また、データタイプレジスタ59は、例外抑止ロード命令の実行においてロードストア対象とされるデータの型 (大きさ) を示す識別値を保持する。そして、レジスタ番号レジスタ61は、例外抑止ロード命令の実行において書き込み対象とするレジスタ番号を保持する。

【0047】

またレジスタタイプレジスタ105は、操作対象とするレジスタの型を示す識別値を保持する。そして、ECレジスタ175は例外抑止ロード命令を実行するとき検出された例外要因の識別値を保持する。ここで、該例外要因とその識別値の例が、以下の表1に示される。

【0048】

【表1】

例外要因	識別値
0 除算	0 x 2 8
データアクセスエラー	0 x 3 2

また、デコード回路 65 は命令実行部 170 から供給された信号を解析し、対応する制御部を起動する。すなわち、供給される登録指示信号に応じて例外抑止履歴登録制御部 183 を起動し、確認指示信号に応じてコミット命令実行部 185 を起動し、読み出し指示信号に応じて例外抑止履歴読み出し命令実行部 187 を起動し、書き込み指示信号に応じて例外抑止履歴書き込み命令実行部 189 を起動する。

【0049】

一方比較器 67～69 は、レジスタ番号レジスタ 61 とレジスタタイプレジスタ 105、及び対応するエントリに接続される。ここで、並設された複数のエントリは例外抑止履歴テーブルを構成するが、各エントリは例外要因を示す EC フィールド (EC) 177～179 と、対応するエントリに例外が発生したか否かを 2 値データにより示す V フィールド 151～153 と、例外情報が格納されたレジスタの種別を示すレジスタタイプフィールド (RT) 107～109 と、例外処理の対象とするデータの実効アドレスを示すアドレスフィールド (ADDR) 71, 75, 79 と、ロード操作の対象とされるデータの型を示すデータタイプフィールド (DT) 72, 76, 80 と、例外情報が書き込まれたレジスタの番号を示すレジスタ番号フィールド (REG #) 73, 77, 81 とを含む。

【0050】

なお、上記データタイプフィールド (DT) 72, 76, 80 にはデータの型に応じた識別値が記録されるが、この識別値の例が以下の表 2 に示される。

【0051】

【表 2】

データ型	識別値
符合なしバイト	0
符合付きバイト	1
符合なしハーフワード	2
符合付きハーフワード	3
ワード	4
ダブルワード	5
クワッドワード	6

上記表 2 に示されるように、該識別値は例えば符号無しバイトに対して 0、符号付きバイトに対しては 1、符号無しハーフワードには 2、符号付きハーフワードには 3、ワードには 4、ダブルワードには 5、クワッドワードには 6 とされる。

【0 0 5 2】

また、上記レジスタタイプフィールド 1 0 7 ～ 1 0 9 には、レジスタの種別に応じた識別値が記録されるが、この識別値の例が以下の表 3 に示される。

【0 0 5 3】

【表 3】

レジスタ種別	識別値
汎用レジスタ	0
浮動小数点レジスタ	1

そして上記比較器 6 7 ～ 6 9 は、レジスタ番号フィールド 7 3, 7 7, 8 1 やレジスタタイプフィールド 1 0 7 ～ 1 0 9 に記録された値から特定される例外情報が書き込まれたレジスタと、コミット命令により指定されたレジスタとが一致

しているか否かを比較し、両者が一致するか否かを示す信号を出力する。

【 0 0 5 4 】

また、コミット判定部 1 8 0 は比較器 6 7 ～ 6 9 に接続され、比較器 6 7 ～ 6 9 から供給された信号に応じて、例外情報が格納されているレジスタがコミット命令により指定されているか否かを判定する。そして、その判定結果をコミット命令実行部 1 8 5 へ出力すると共に、コミット信号 CM をコミット例外割り込み制御部 4 4 へ供給する。

【 0 0 5 5 】

そして、コミットエントリ検出部 1 8 1 は、比較器 6 7 ～ 6 9 に接続され、比較器 6 7 ～ 6 9 から供給された信号に応じて、コミット命令により指定されたレジスタの番号と所定のフィールドに記録されているレジスタの番号とが一致したエントリの番号を検出する。

【 0 0 5 6 】

また、無効エントリ検出部 8 7 は、各エントリの EC フィールド 1 7 7, 1 7 8, 1 7 9 の情報に応じて、空きエントリ（無効なエントリ）の検出を行う。また、例外抑止履歴登録制御部 1 8 3 は、アドレスレジスタ 5 7 とデータタイプレジスタ 5 9、レジスタ番号レジスタ 6 1、レジスタタイプレジスタ 1 0 5、EC レジスタ 1 7 5、デコード回路 6 5 及び無効エントリ検出部 8 7 に接続され、デコード回路 6 5 から供給された登録信号 ADD に応じて、無効エントリ検出部 8 7 で検出された空きエントリのアドレスフィールド (ADDR) 7 1, 7 5, 7 9 とレジスタタイプフィールド 1 0 7, 1 0 8, 1 0 9 と、レジスタ番号フィールド (REG #) 7 3, 7 7, 8 1 へ例外情報を書き込む。

【 0 0 5 7 】

また、コミット命令実行部 1 8 5 は、レジスタ番号レジスタ 6 1 と、レジスタタイプレジスタ 1 0 5 と、コミット判定部 1 8 0 と、コミットエントリ検出部 1 8 1 と、デコード回路 6 5 及び V フィールド 1 5 1 ～ 1 5 3 に接続される。そして、コミット命令実行部 1 8 5 は、コミット判定部 1 8 0 においてコミット命令により指定されたレジスタの番号と所定のフィールドに記録されているレジスタの番号とが一致したと判定された場合、一致した該レジスタ番号が格納されてい

るレジスタのVフィールド151～153へ1を書き込む。

【0058】

また、例外抑止履歴読みだし命令実行部187は、指定されたエントリから例外情報を読み出して命令実行部170へ供給する。さらに、例外抑止履歴書き込み命令実行部189は、命令実行部170から供給された値を指定されたエントリへ書き込む。

【0059】

次に、上記の計算機に係る動作の概要を説明する。なお、初期状態は通常動作を行うものとする。まず、該通常の動作においては命令読み出し部3は、プログラムカウンタ（PC）13が示す命令語を読み出し、命令実行部170へ供給する。そして、命令実行部170は供給された命令を実行するが、例外抑止ロード命令が供給された場合には、該実行時に例外処理の必要性を検出した場合において、書き込み対象とするレジスタに対応する例外抑止フラグ38を1と設定して有効にすると共に、履歴制御部173に例外情報を登録する。

【0060】

そして、コミット命令が供給された場合には、汎用レジスタ37内の指定されたレジスタの番号に対応する例外抑止フラグ38が有効か否か確認し、有効な場合には履歴制御部173を通じて抑止されていたコミット例外を発生させる。なお、通常動作においては、以上のような動作を繰り返す。

【0061】

また、割り込みが発生した場合には、命令読み出し部3又は命令実行部170から供給された割込信号に応じて、割込制御回路40は割り込みから復帰する先の命令語アドレスをレジスタ31へ、割り込み発生前の動作状態をレジスタ33へ、発生した割り込みに対応する動作状態をレジスタ35へ書き込む。また、発生した割り込みに対応する分岐先アドレスをプログラムカウンタ13へ供給する。ここで命令読み出し部3は、割り込み制御部10から供給された分岐先アドレスに応じてメモリ1から命令語を読み出し、該命令語を命令実行部170へ供給する。そして、以降においては上記の通常の動作と同様である。

【0062】

また、割り込みから復帰する場合には、命令実行部 1 7 0 は割り込み復帰命令を実行することによりレジスタ 3 3 に格納された値をレジスタ 3 5 へ書き込む。また、レジスタ 3 1 に格納された値を読み出し、分岐先アドレスとして命令読み出し部 3 へ供給する。ここで、命令読み出し部 3 は供給された分岐先アドレスに応じて命令語をメモリ 1 から読み出し、命令実行部 1 7 0 へ供給する。そして、以降においては上記の通常動作と同様に動作する。

【 0 0 6 3 】

次に、図 1 2 を参照しつつ、本実施の形態 1 に係る計算機において、コミット例外における割り込み処理プログラムを実行する手順について説明する。図 1 2 に示されるように、ステップ S 1 ではコンテキストを退避する。次にステップ S 2 において、例外抑止履歴テーブル内で V フィールドが有効なエントリを求める。そしてステップ S 3 では、有効なエントリが見出せたか否か判断し、見出せない場合にはステップ S 2 0 へ進み、不当なコミット例外としてエラーと処理する。なお、この場合には次にステップ S 8 へ進む。

【 0 0 6 4 】

一方、ステップ S 3 において有効なエントリが見出せた場合には、ステップ S 4 へ進み、E C フィールドに記録されたデータに基づいて回復可能な例外処理か否かを特定する。そしてステップ S 5 において、該例外処理が回復可能なものであるか否かを判断し、回復可能でない場合にはステップ S 1 0 へ進み、異常終了処理がなされる。また、ステップ S 5 において回復可能と判断された場合には、ステップ S 6 へ進み、例外処理が実行される。そして、ステップ S 7 において、例外抑止履歴書き込み命令により、例外抑止履歴テーブルのエントリを無効化する。

【 0 0 6 5 】

次に、ステップ S 8 において、本来の処理におけるコンテキストが復元され、ステップ S 9 で、割り込み復帰命令によりコミット例外から復帰する。

【 0 0 6 6 】

以上より、本発明の実施の形態 1 に係る計算機によれば、命令の実行順序に起因して例外処理を留保する必要がある場合には、該例外処理を実行するために必

要な例外情報が履歴制御部 1 7 3 に格納され、留保した例外処理を実行すべきであることが本来の処理で確定したときに初めて例外処理が実行される。そしてこのとき、上記のように、履歴制御部 1 7 3 に予め格納された上記例外情報に応じて該例外処理がなされ、その後本来の処理が続行されるため、動作速度が向上される。

〔実施の形態 2〕

図 1 3 は、本発明の実施の形態 2 に係る計算機の構成を示す図である。図 1 3 に示されるように、本実施の形態 2 に係る計算機は、図 7 に示された実施の形態 1 に係る計算機と同様な構成を有するが、命令実行部 1 9 0 には浮動小数点ロード命令実行部 2 5 と、浮動小数点ストア命令実行部 2 7 と、浮動小数点演算命令実行部 2 9 と、浮動小数点コミット命令実行部 1 9 1 と、例外抑止・浮動小数点演算命令実行部 1 9 9 とを含み、履歴制御部 2 0 7 の構成が変更される点で相違するものである。また、レジスタ制御部 2 0 3 には浮動小数点レジスタ 3 9 と例外抑止フラグ 2 0 5 とがさらに備えられる点でも相違する。

【0 0 6 7】

ここで、浮動小数点ロード命令実行部 2 5 はメモリ 1 及び命令デコード部 1 7 と汎用レジスタ 3 7 に接続され、浮動小数点ストア命令実行部 2 7 及び浮動小数点演算命令実行部 2 9 は命令デコード部 1 7 及び浮動小数点レジスタ 3 9 に接続される。そして、さらに浮動小数点ストア命令実行部 2 7 は汎用レジスタ 3 7 に接続され、浮動小数点演算命令実行部 2 9 は条件レジスタ 3 0 に接続される。なお、浮動小数点ロード命令実行部 2 5 と浮動小数点ストア命令実行部 2 7 及び浮動小数点演算命令実行部 2 9 は共に割込制御回路 4 0 に接続される。

【0 0 6 8】

また、浮動小数点コミット命令実行部 1 9 1 は命令デコード部 1 7 と、例外抑止・浮動小数点演算命令実行部 1 9 9 と、履歴制御部 2 0 7 と、浮動小数点ストア命令実行部 2 7 及び浮動少数点レジスタ 3 9 に接続される。そして、例外抑止・浮動小数点演算命令実行部 1 9 9 はさらに、命令デコード部 1 7 と、浮動少数点レジスタ 3 9 と、履歴制御部 2 0 7 に接続される。

【0 0 6 9】

ここで、浮動小数点ロード命令実行部 2 5 は、浮動小数点ロード命令が供給された場合には、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応するメモリ 1 の領域からデータを読み出し、その結果を浮動小数点レジスタ 3 9 へ書き込む。なお、浮動小数点ロード命令を実行する際に割り込みを検出した場合には、割り込信号を割り込み制御回路 4 0 へ供給する。

【 0 0 7 0 】

また、浮動小数点ストア命令実行部 2 7 は、浮動小数点ストア命令が供給された場合には、汎用レジスタ 3 7 から読み出された値に基づいて求められた実効アドレスに対応する浮動小数点レジスタ 3 9 の領域からデータを読み出し、その結果を実効アドレスに対応するメモリ 1 の領域へ書き込む。なお、浮動小数点ストア命令を実行する際に割り込みを検出した場合には、割り込信号を割り込み制御回路 4 0 へ供給する。また、浮動小数点演算命令実行部 2 9 は、浮動小数点演算命令が供給された場合には浮動小数点レジスタ 3 9 から読み出した値に基づき演算を実行し、その結果を浮動小数点レジスタ 3 9 へ書き込む。

【 0 0 7 1 】

次に、例外抑止・浮動小数点演算命令実行部 1 9 9 は、命令デコード部 1 7 より例外抑止・浮動小数点演算命令が供給された場合には、浮動小数点レジスタ 3 9 から読み出した値に基づいて浮動小数点演算を行い、その結果を浮動小数点レジスタ 3 9 へ書き込む。なお、命令を実行する際に割り込みを検出した場合には、浮動小数点レジスタ 3 9 内の書き込みレジスタの番号に対応する例外抑止フラグ 2 0 5 を 1 に設定する。また、履歴制御部 2 0 7 に対して登録指示信号を供給することにより例外情報を履歴制御部 2 0 7 に記憶する。

【 0 0 7 2 】

そして、浮動小数点コミット命令実行部 1 9 1 は命令デコード部 1 7 より浮動小数点コミット命令が供給された場合には、浮動小数点レジスタ 3 9 内の指定されたレジスタの番号に対応する例外抑止フラグ 2 0 5 が 1 であれば、該フラグを 0 とし、履歴制御部 2 0 7 に対して確認指示信号を供給する。これにより、履歴制御部 2 0 7 からコミット信号 CM が割り込み制御部 1 0 へ供給され、コミット例外の発生が割り込み制御部 1 0 へ伝えられる。なお、上記例外抑止フラグ 2 0

5は、その各ビットが浮動小数点レジスタ39内の各エントリに対応して設けられる。

【0073】

図14は、図13に示された履歴制御部207の構成を示す図である。図14に示されるように、履歴制御部207は図8に示された履歴制御部173と同様な構成を有するが、命令語レジスタ213をさらに備え、各エントリには例外処理の起因となる例外起因命令の命令語を示す命令語フィールド209～211が含まれる点で相違する。

【0074】

ここで、命令語レジスタ213の入力端は命令実行部170に接続され、出力端は例外抑止履歴登録制御部183と例外抑止履歴書き込み命令実行部189に接続される。上記において、命令語レジスタ213が、命令実行部170から供給された例外起因命令の命令語を保持すると共に、命令語フィールド209～211には該命令語を示す値が、例外抑止履歴書き込み命令実行部189により書き込まれる。

【0075】

そして、該命令語フィールド209～211に書き込まれた命令語を示す値は、例外抑止履歴読み出し命令実行部187により読み出され、命令実行部170に供給される。

【0076】

ここで、本実施の形態2に係る例外抑止・浮動小数点演算命令による動作を、図15を参照しつつ説明する。図15に示されるように、ステップS1において読み出したデータの中に例外要因が存在するか否か確認する。その結果、ステップS2において例外要因が存在しないと判断された場合には、ステップS3へ進み、読み出した上記データに基づき演算を行い、その結果を浮動小数点レジスタ39へ書き込む。

【0077】

一方、ステップS2において例外要因が存在すると判断された場合には、ステップS10へ進み、履歴制御部207へ登録指示信号が供給されることにより、

検出された命令語等の例外情報が履歴制御部 2 0 7 に登録される。このとき、ステップ S 1 1 において、上記例外情報が書き込まれるレジスタに対応する例外抑止フラグ 2 0 5 に 1 の値が設定され有効とされる。

【 0 0 7 8 】

以上より、本実施の形態 2 に係る計算機によれば、例外情報として例外起因命令語も保持され、例外処理を終えた後に上記例外起因命令から本来の処理が順次再開されるため、上記実施の形態 1 に係る計算機と同様な効果を奏すると共に、動作の信頼性を高めることができる。

〔実施の形態 3〕

図 1 6 は、本発明の実施の形態 3 に係る計算機の構成を示す図である。図 1 6 に示されるように、本実施の形態 3 に係る計算機は、図 7 に示された実施の形態 1 に係る計算機と同様な構成を有するが、命令実行部 2 1 5 には例外抑止履歴無効化制御部 2 1 7 をさらに含み、履歴制御部 2 1 9 には図 1 7 に示されるように、履歴テーブル無効化部 2 1 8 を含む点で相違するものである。

【 0 0 7 9 】

ここで、上記例外抑止履歴無効化制御部 2 1 7 は入力端が命令デコード部 1 7 に接続され、出力端が汎用レジスタ 3 7 及び履歴制御部 2 1 9 に接続される。

【 0 0 8 0 】

そして、この例外抑止履歴無効化制御部 2 1 7 は、例外抑止フラグ無効化命令が命令デコード部 1 7 から供給された場合には、図 1 8 のステップ S 1 に示されるように、指定されたレジスタに対応する例外抑止フラグ 3 8 を無効化する。そしてさらに、ステップ S 2 において、指定されたレジスタ番号に対応する例外情報が保持されている場合には、該例外情報が保持されているエントリを無効化するためのリセット信号 R S を履歴制御部 2 1 9 へ供給する。

【 0 0 8 1 】

ここで、履歴制御部 2 1 9 に含まれた履歴テーブル無効化部 2 1 8 は、該リセット信号 R S に応じて、指定されたレジスタに対応する例外情報を保持する履歴テーブルのエントリを、その E C フィールドを無効化することにより無効化する。

【 0 0 8 2 】

以上より、本実施の形態 3 に係る計算機によれば、上記実施の形態 1 に係る計算機と同様な効果を奏すると共に、例外抑止フラグ無効化命令を実行することにより、例外抑止フラグを無効化することができるため、複数の投機的命令移動を可能として、さらに動作の高速化を図ることができる。

〔実施の形態 4〕

図 1 9 は、本発明の実施の形態 4 に係る計算機の構成を示す図である。図 1 9 に示されるように、本実施の形態 4 に係る計算機は、図 7 に示された実施の形態 1 に係る計算機と同様な構成を有するが、命令実行部 2 2 0 には例外抑止フラグ読み出し命令実行部 2 2 1 と例外抑止フラグ書き込み命令実行部 2 2 3 をさらに含む点で相違するものである。

【 0 0 8 3 】

ここで、例外抑止フラグ読み出し命令実行部 2 2 1 と例外抑止フラグ書き込み命令実行部 2 2 3 とは共に、入力端が命令デコード部 1 7 と汎用レジスタ 3 7 に接続され、出力端が汎用レジスタ 3 7 に接続される。

【 0 0 8 4 】

そして、例外抑止フラグ読み出し命令実行部 2 2 1 は、命令デコード部 1 7 から供給された例外抑止フラグ読み出し命令に応じて例外抑止フラグ 3 8 の値を読み出し、汎用レジスタ 3 7 へ書き込む。また、例外抑止フラグ書き込み命令実行部 2 2 3 は、命令デコード部 1 7 から供給された例外抑止フラグ書き込み命令に応じて汎用レジスタ 3 7 からデータを読み出し、それを例外抑止フラグ 3 8 へ書き込む。

【 0 0 8 5 】

以上のような本実施の形態 4 に係る計算機によれば、上記実施の形態 1 に係る計算機と同様な効果を奏すると共に、例外抑止フラグ 3 8 の値を汎用レジスタ 3 7 へ退避することができるため、重畳的な投機的命令移動を可能として、さらに動作の高速化を図ることができる。

〔実施の形態 5〕

本発明の実施の形態 5 に係る計算機は、図 7 に示された実施の形態 1 に係る計

算機と同様な構成を有するが、履歴制御部の構成が相違するものである。ここで図 20 は、本発明の実施の形態 5 に係る履歴制御部 225 の構成を示す図である。図 20 に示されるように、本実施の形態に係る履歴制御部 225 は、図 8 に示された実施の形態 1 に係る履歴制御部 173 と同様な構成を有するが、例外 PC レジスタ (EPC レジスタ) 227 をさらに備え、例外抑止履歴テーブルを構成する各エントリには EPC フィールド 229 ~ 231 が設けられる点で相違する。

【0086】

ここで、例外 PC レジスタ 227 は入力端が命令実行部 170 に接続され、出力端が例外抑止履歴登録制御部 183 と例外抑止履歴書き込み命令実行部 189 に接続される。そして、例外 PC レジスタ 227 は例外起因命令の一つである例外抑止ロード命令の命令アドレスを保持する。また、各 EPC フィールド 229 ~ 231 には、例外起因命令の命令アドレスが記録される。

【0087】

次に、図 21 を参照しつつ、本実施の形態 5 に係る計算機において、コミット例外における割り込み処理プログラムを実行する手順について説明する。図 21 に示されるように、ステップ S1 ではコンテキストを退避する。次にステップ S2 において、例外抑止履歴テーブル内で V フィールドが有効なエントリを求める。そしてステップ S3 では、有効なエントリが見出せたか否か判断し、見出せない場合にはステップ S30 へ進み、不当なコミット例外としてエラーと処理する。なお、この場合には次にステップ S9 へ進む。

【0088】

一方、ステップ S3 において有効なエントリが見出せた場合には、ステップ S4 へ進み、EC フィールドに記録されたデータに基づいて例外要因が解消可能か否かを特定する。そしてステップ S5 において、該例外要因が解消可能なものであるか否かを判断し、解消可能でない場合にはステップ S20 へ進み、異常終了処理がなされる。なお、この場合には次にステップ S7 へ進む。

【0089】

また、ステップ S5 において解消可能と判断された場合には、ステップ S6 へ

進み、例外要因の解消処理が実行される。そして、ステップ S 7 において、例外起因命令の命令アドレスをコンテキスト復元時のレジスタ (EPCR) 3 1 の値として設定する。次に、ステップ S 8 において、例外抑止履歴書き込み命令により、例外抑止履歴テーブルのエントリを無効化する。

【0090】

そして、ステップ S 9 において、本来の処理におけるコンテキストが復元され、ステップ S 10 で、割り込み復帰命令によりコミット例外から復帰する。

【0091】

以上のように、本実施の形態 5 に係る計算機によれば、実施の形態 1 に係る計算機と同様の効果を奏すると共に、例外抑止履歴テーブルへ例外起因命令の命令アドレスが保持されるため、例外要因を解消した後に例外起因命令に戻って、本来の処理を確実に続行することができる。

〔実施の形態 6〕

図 2 2 は、本発明の実施の形態 6 に係る計算機の構成を示す図である。図 2 2 に示されるように、本実施の形態 6 に係る計算機は、図 7 に示された実施の形態 1 に係る計算機と同様な構成を有するが、命令実行部 2 3 9 に含まれるロード命令実行部 2 3 3 と演算命令実行部 2 3 5、及び例外抑止ロード命令実行部 2 3 7 の構成が相違するものである。

【0092】

図 2 3 は、図 2 2 に示されたロード命令実行部 2 3 3 の構成を示す図である。図 2 3 に示されるように、ロード命令実行部 2 3 3 は、ロード制御回路 2 4 1 と、フラグ検出回路 2 4 3 と、セレクタ 2 4 5 とを含む。ここで、ロード制御回路 2 4 1 の入力端は命令デコード部 1 7 と、メモリ 1 及び汎用レジスタ 3 7 に接続され、出力端は割込制御回路 4 0 と、汎用レジスタ 3 7 及びセレクタ 2 4 5 に接続される。また、フラグ検出回路 2 4 3 の入力端は汎用レジスタ 3 7 に接続され、出力端はセレクタ 2 4 5 に接続される。さらに、セレクタ 2 4 5 の出力端は汎用レジスタ 3 7 に接続される。

【0093】

このような構成を有するロード命令実行部 2 3 3 においては、図 2 5 に示され

るように、ステップ S 1 において G R 読出データに応じて、フラグ検出回路 2 4 3 によりデータの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有効か否かが確認され、ステップ S 2 において無効であると判定された場合には、ステップ S 3 へ進み、ロード制御回路 2 4 1 から供給されたデータが、G R 書き込みデータとしてセクタ 2 4 5 から選択的に汎用レジスタ 3 7 へ出力される。

【 0 0 9 4 】

なお、上記において G R 読出データは、汎用レジスタ 3 7 から読み出されたデータと該データに対応する例外抑止フラグ 3 8 の値とからなり、以下において同様である。

【 0 0 9 5 】

一方、ステップ S 2 においてフラグ検出回路 2 4 3 により、データの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有効であると判定された場合には、ステップ S 1 0 へ進み、フラグ検出回路 2 4 3 から供給され書き込み先のレジスタに対応する例外抑止フラグ 3 8 を有効とする信号が、セクタ 2 4 5 から選択的に出力される。

【 0 0 9 6 】

図 2 4 は、図 2 2 に示された演算命令実行部 2 3 5 の構成を示す図である。図 2 4 に示されるように、演算命令実行部 2 3 5 は、演算制御回路 2 4 7 と、フラグ検出回路 2 4 3 と、セクタ 2 4 5 とを含む。ここで、演算制御回路 2 4 7 の入力端は命令デコード部 1 7 及び汎用レジスタ 3 7 に接続され、出力端は割込制御回路 4 0 と、汎用レジスタ 3 7 及びセクタ 2 4 5 に接続される。また、フラグ検出回路 2 4 3 の入力端は汎用レジスタ 3 7 に接続され、出力端はセクタ 2 4 5 に接続される。さらに、セクタ 2 4 5 の出力端は汎用レジスタ 3 7 に接続される。

【 0 0 9 7 】

このような構成を有する演算命令実行部 2 3 5 においては、図 2 6 に示されるように、ステップ S 1 において G R 読出データに応じて、フラグ検出回路 2 4 3 によりデータの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有

効か否かが確認され、ステップ S 2 において無効であると判定された場合には、ステップ S 3 へ進み、演算制御回路 2 4 7 から供給されたデータが、G R 書き込みデータとしてセクタ 2 4 5 から選択的に汎用レジスタ 3 7 へ出力される。

【 0 0 9 8 】

一方、ステップ S 2 においてフラグ検出回路 2 4 3 により、データの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有効であると判定された場合には、ステップ S 1 0 へ進み、フラグ検出回路 2 4 3 から供給され書き込み先のレジスタに対応する例外抑止フラグ 3 8 を有効とする信号が、セクタ 2 4 5 から選択的に出力される。

【 0 0 9 9 】

なお、本実施の形態に係る例外抑止ロード命令実行部 2 3 7 の構成及び動作は、上記ロード命令実行部 2 3 3 や演算命令実行部 2 3 5 の構成及び動作と同様であるが、以下において本実施の形態 6 に係る例外抑止ロード命令実行部 2 3 7 の動作を、図 2 7 のフローチャートを参照しつつ説明する。

【 0 1 0 0 】

図 2 7 に示されるように、ステップ S 1 において G R 読出データに応じて、データの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有効か否かが確認され、ステップ S 2 において無効であると判定された場合には、ステップ S 3 へ進み、読み出したデータの中に例外要因が存在するか否か確認する。

【 0 1 0 1 】

一方、ステップ S 2 において、データの読み出し対象としたレジスタに対応する例外抑止フラグ 3 8 が有効であると判定された場合には、ステップ S 2 0 へ進み、書き込み先のレジスタに対応する例外抑止フラグ 3 8 を有効とする信号が出力される。

【 0 1 0 2 】

ステップ S 4 では、例外要因が存在しないと判断された場合には、ステップ S 5 へ進みロード対象アドレス域から読み出した上記データを、汎用レジスタ 3 7 へ書き込む。

【 0 1 0 3 】

一方、ステップ S 4 において例外要因が存在すると判断された場合には、ステップ S 1 0 へ進み、履歴制御部 1 7 3 へ登録指示信号が供給されることにより、検出された例外情報が履歴制御部 1 7 3 に登録される。このとき、ステップ S 1 1 において、上記例外情報が記憶されるレジスタに対応する例外抑止フラグ 3 8 に 1 の値が設定され有効とされる。

【 0 1 0 4 】

以上より、本実施の形態 6 に係る計算機によれば、命令の実行に際して、データ読み出し先のレジスタに対応する例外抑止フラグ 3 8 が有効である場合には、実行結果の書き込み先となるレジスタに対応する例外抑止フラグ 3 8 も有効にするため、例外抑止フラグ 3 8 の有効情報を伝播させることができ、投機的に移動した命令と依存関係を持った命令も移動することができる。これより、投機的な命令の移動についての自由度を高めることができ、動作速度をさらに高めることができる。

【 0 1 0 5 】

最後に、本発明の課題を解決するための手段について付記する。

(1) プログラムされた命令の実行を制御する計算機の制御方法であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行する第一のステップと、第一のステップにおいて、例外動作の必要性が検出された場合には、例外動作を留保する第二のステップと、分岐命令を実行することにより選択された分岐先の命令の実行において、第二のステップで留保した例外動作が必要とされる場合には、例外動作を遂行する第三のステップと、例外動作が終了したときには、プログラムに復帰して分岐先の命令の実行を続行する第四のステップとを有することを特徴とする計算機の制御方法。

(2) プログラムされた命令の実行を制御する計算機の制御方法であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行する第一のステップと、第一のステップにおいて、例外動作を必要とする例外起因命令が検出された場合には、例外動作を留保する第二のステップと、分岐命令を実行することにより選択された分岐先の命令の実行において、第二のステップで留保した例外動作が必要とされる場合には、例外動作を遂行する第三のステップと、例

外動作が終了したときには、プログラムに復帰して例外起因命令から順次実行する第四のステップとを有することを特徴とする計算機の制御方法。

(3) 第三のステップにおいては、例外動作が割り込み処理プログラムを実行することにより遂行される(1)または(2)のいずれかに記載の計算機の制御方法。

(4) 第二のステップにおいては、留保される例外動作を実行するための情報が記憶され、第三のステップにおいては、情報に応じて例外動作が遂行される(1)または(2)のいずれかに記載の計算機の制御方法。このように、留保される例外動作を実行するための情報が記憶され、該情報に応じて例外動作が遂行されるようにすれば、例外動作を実行するために必要なプログラムを削減することができるため、計算機の回路規模を低減することができると共に、動作の高速化を図ることができる。

(5) 第二のステップでは、例外動作による処理が必要とされるデータか否かを示す識別情報が、所定の処理の結果得られるデータ毎に付与され、第三のステップでは、分岐命令を実行することにより選択された分岐先の命令を実行する場合において、識別情報に応じて例外動作による処理が必要であると判別されるデータが実行対象とされるとき、例外動作が遂行される(1)または(2)のいずれかに記載の計算機の制御方法。このように、例外動作による処理が必要とされるデータか否かを示す識別情報が、所定の処理の結果得られるデータ毎に付与され、該識別情報に応じて例外動作による処理が必要であるか否かが判別されることとすれば、さらに動作の高速化を図ることができる。

(6) 所定の命令を実行することにより、留保される例外動作を実行するための情報が無効化される(4)に記載の計算機の制御方法。

(7) 所定の命令を実行することにより、識別情報が無効化される(5)に記載の計算機の制御方法。このように、所定の命令を実行することにより、留保される例外動作を実行するための情報や識別情報を無効化できるようにすれば、複数の命令について実行順序を変更することができるため、計算機の処理能力を高めることができる。

(8) 所定の命令を実行することにより、識別情報を読み出し、又は識別情報を

書き換える（５）に記載の計算機の制御方法。このように、所定の命令を実行することによって、識別情報を読み出し、又は識別情報を書き換えることができるようにすれば、複数の命令において重疊的に実行順序を変更することができるため、計算機の処理能力を高めることができる。

（９）プログラムされた命令を実行する計算機であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行するよう制御する制御手段と、後置された命令の実行中に、例外動作の必要性が検出された場合には、例外動作を留保する例外抑止手段と、分岐命令を実行することにより選択された分岐先の命令の実行において、例外抑止手段により留保された例外動作が必要とされる場合には、例外動作を遂行する例外動作実行手段と、例外動作が終了したときには、プログラムに復帰して分岐先の命令の実行を続行する復帰手段とを備えたことを特徴とする計算機。

（１０）プログラムされた命令を実行する計算機であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行するよう制御する制御手段と、後置された命令の実行中に、例外動作を必要とする例外起因命令が検出された場合には、例外動作を留保する例外抑止手段と、分岐命令を実行することにより選択された分岐先の命令の実行において、例外抑止手段により留保された例外動作が必要とされる場合には、例外動作を遂行する例外動作実行手段と、例外動作が終了したときには、プログラムに復帰して例外起因命令から順次実行する復帰手段とを備えたことを特徴とする計算機。

（１１）例外動作実行手段は、割り込み処理プログラムを実行することにより例外動作を遂行する（９）または（１０）のいずれかに記載の計算機。

（１２）例外抑止手段により留保される例外動作を実行するための情報を記憶する記憶手段をさらに備え、例外動作実行手段は、記憶手段に記憶された情報に応じて例外動作を遂行する（９）または（１０）のいずれかに記載の計算機。

（１３）例外抑止手段は、例外動作による処理が必要とされるか否かを示す識別情報を、所定の処理の結果得られるデータ毎に付与し、例外動作実行手段は、分岐命令を実行することにより選択された分岐先の命令を実行する場合において、識別情報に応じて例外動作による処理が必要であると判別されるデータが実行対

象とされるときに、例外動作を遂行する（９）または（１０）のいずれかに記載の計算機。

（１４）所定の命令を実行することにより、留保された例外動作を実行するための情報を無効化する履歴無効化手段をさらに備えた（１２）に記載の計算機。

（１５）所定の命令を実行することにより、識別情報を無効化する識別情報無効化手段をさらに備えた（１３）に記載の計算機。

（１６）所定の命令を実行することにより、識別情報を読み出す識別情報読み出し手段と、所定の命令を実行することにより、識別情報を書き換える識別情報書き換え手段とをさらに備えた（１３）に記載の計算機。

【 0 1 0 6 】

【発明の効果】

上述の如く、プログラムにおいて分岐命令より後置された命令を該分岐命令より先に実行したために留保された例外動作の実行が必要とされる場合に、例外動作を遂行すると共に、例外動作が終了したときにはプログラムに復帰して例外動作の遂行が必要とされた命令の次の命令から順次実行することにより、動作の高速化を図ることができるため、計算機の処理能力を高めることができる。

【 0 1 0 7 】

また、例外動作が終了したときには、プログラムに復帰して例外動作の起因となる命令から順次実行することによっても、動作の高速化を図ることができるため、計算機の処理能力を高めることができる。

【図面の簡単な説明】

【図 1】

汎用レジスタと浮動小数点レジスタとを有する従来の計算機の構成を示す図である。

【図 2】

ロード命令による動作を示すフローチャートである。

【図 3】

ロード命令の命令形式を示す図である。

【図 4】

ストア命令による動作を示すフローチャートである。

【図 5】

ストア命令の命令形式を示す図である。

【図 6】

演算命令による動作を示すフローチャートである。

【図 7】

本発明の実施の形態 1 に係る計算機の構成を示す図である。

【図 8】

図 7 に示された履歴制御部の構成を示す図である。

【図 9】

実施の形態 1 に係る例外抑止ロード命令による動作を示すフローチャートである。

【図 1 0】

実施の形態 1 に係るコミット命令による動作を示すフローチャートである。

【図 1 1】

実施の形態 1 に係るコミット命令の命令形式を示す図である。

【図 1 2】

実施の形態 1 に係るコミット例外における割り込み処理を示すフローチャートである。

【図 1 3】

本発明の実施の形態 2 に係る計算機の構成を示す図である。

【図 1 4】

図 1 3 に示された履歴制御部の構成を示す図である。

【図 1 5】

実施の形態 2 に係る例外抑止浮動小数点演算命令による動作を示すフローチャートである。

【図 1 6】

本発明の実施の形態 3 に係る計算機の構成を示す図である。

【図 1 7】

図 1 6 に示された履歴制御部の構成を示す図である。

【図 1 8】

実施の形態 3 に係る例外抑止フラグ無効化命令による動作を示す図である。

【図 1 9】

本発明の実施の形態 4 に係る計算機の構成を示す図である。

【図 2 0】

本発明の実施の形態 5 に係る履歴制御部の構成を示す図である。

【図 2 1】

実施の形態 5 に係るコミット例外における割り込み処理を示すフローチャートである。

【図 2 2】

本発明の実施の形態 6 に係る計算機の構成を示す図である。

【図 2 3】

図 2 2 に示されたロード命令実行部の構成を示す図である。

【図 2 4】

図 2 2 に示された演算命令実行部の構成を示す図である。

【図 2 5】

実施の形態 6 に係るロード命令による動作を示すフローチャートである。

【図 2 6】

実施の形態 6 に係る演算命令による動作を示すフローチャートである。

【図 2 7】

実施の形態 6 に係る例外抑止ロード命令による動作を示すフローチャートである。

【符号の説明】

- 1 メモリ
- 3 命令読み出し部
- 6, 170, 190, 215, 220, 239 命令実行部
- 8, 171, 203 レジスタ制御部
- 9, 10 割り込み制御部

- 1 1 命令読み出し制御部
- 1 3 プログラムカウンタ (P C)
- 1 5 命令語レジスタ (I R)
- 1 7 命令デコード部
- 1 9, 2 3 3 ロード命令実行部
- 2 0 例外抑止履歴書き込み命令実行部
- 2 1 ストア命令実行部
- 2 2, 2 3 5 演算命令実行部
- 2 3 命令実行回路
- 2 4, 2 3 7 例外抑止ロード命令実行部
- 2 5 浮動小数点 (F) ロード命令実行部
- 2 6 コミット命令実行部
- 2 7 浮動小数点 (F) ストア命令実行部
- 2 8 例外抑止履歴読出し命令実行部
- 2 9 浮動小数点 (F) 演算命令実行部
- 3 0 条件レジスタ
- 3 1 レジスタ (E P C R)
- 3 3 レジスタ (E P S R)
- 3 5 レジスタ (P S R)
- 3 7 汎用レジスタ
- 3 8, 2 0 5 例外抑止フラグ
- 3 9 浮動小数点レジスタ
- 4 0 割込制御回路
- 4 4 コミット例外割り込み制御部
- 1 7 3, 2 0 7, 2 1 9, 2 2 5 履歴制御部
- 5 7 アドレスレジスタ
- 5 9 データタイプレジスタ
- 6 1 レジスタ番号レジスタ
- 6 5 デコード回路

- 6 7 ~ 6 9 比較器
- 7 1, 7 5, 7 9 アドレスフィールド (ADDR)
- 7 2, 7 6, 8 0 データタイプフィールド (DT)
- 7 3, 7 7, 8 1 レジスタ番号フィールド (REG #)
- 8 7 無効エントリ検出部
- 1 0 5 レジスタタイプレジスタ (RTREG)
- 1 0 7 ~ 1 0 9 レジスタタイプフィールド
- 1 5 1 ~ 1 5 3, V0 ~ Vm Vフィールド
- 1 7 5 例外要因レジスタ (ECレジスタ)
- 1 7 7 ~ 1 7 9 ECフィールド
- 1 8 0 コミット判定部
- 1 8 1 コミットエントリ検出部
- 1 8 3 例外抑止履歴登録制御部
- 1 8 5 コミット命令実行部
- 1 8 7 例外抑止履歴読み出し命令実行部
- 1 8 9 例外抑止履歴書き込み命令実行部
- 1 9 1 浮動小数点コミット命令実行部
- 1 9 9 例外抑止・浮動小数点演算命令実行部
- 2 0 9 ~ 2 1 1 命令語フィールド (INST)
- 2 1 3 命令語レジスタ
- 2 1 7 例外抑止履歴無効化制御部
- 2 1 8 履歴テーブル無効化部
- 2 2 1 例外抑止フラグ読み出し命令実行部
- 2 2 3 例外抑止フラグ書き込み命令実行部
- 2 2 7 例外PCレジスタ (EPCレジスタ)
- 2 2 9 ~ 2 3 1 EPCフィールド
- 2 4 1 ロード制御回路
- 2 4 3 フラグ検出回路
- 2 4 5 セレクタ

2 4 7 演算制御回路

A D D 登録信号

C C 確認信号

R 読み出し信号

W 書き込み信号

C M コミット信号

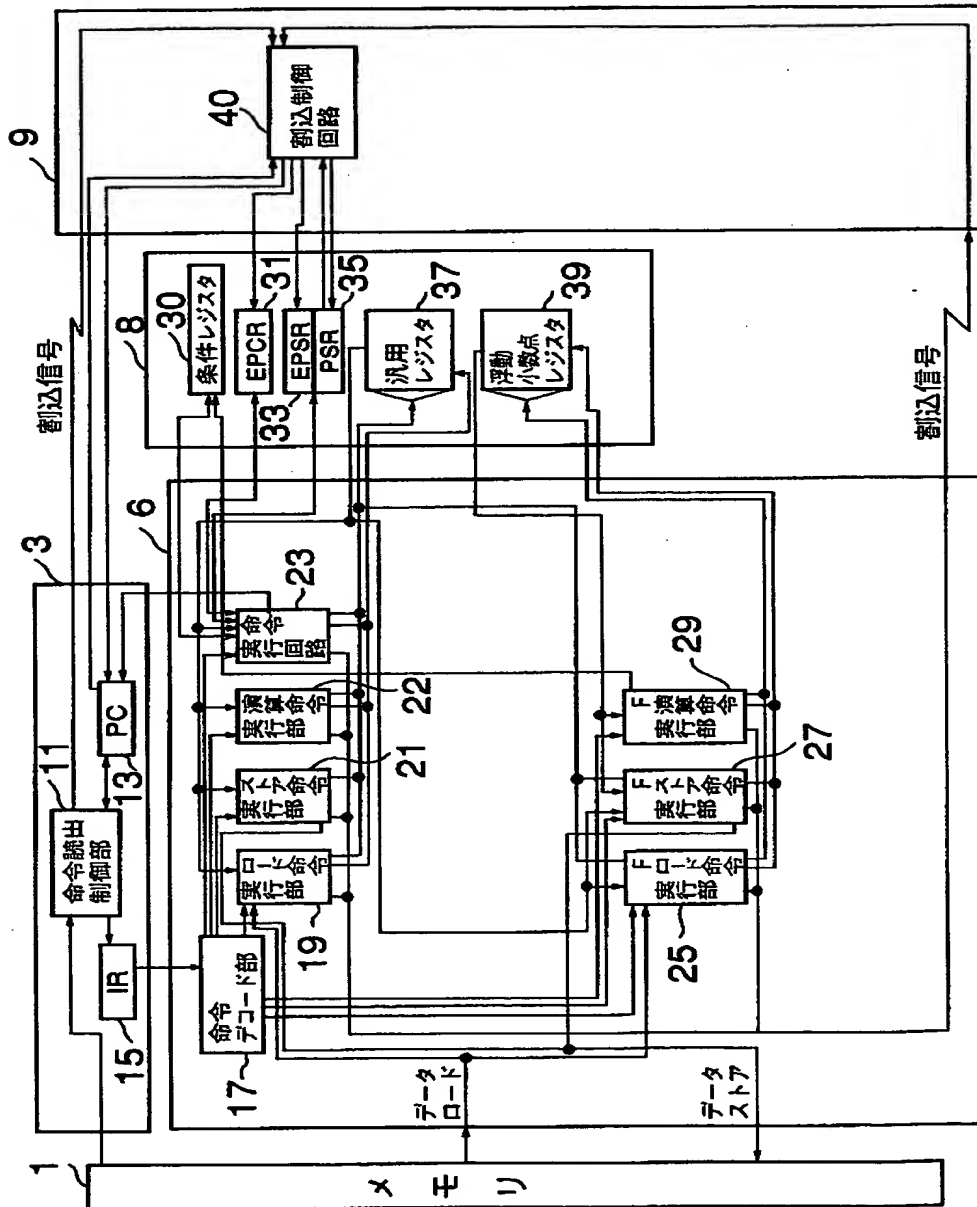
R S リセット信号

【書類名】

図面

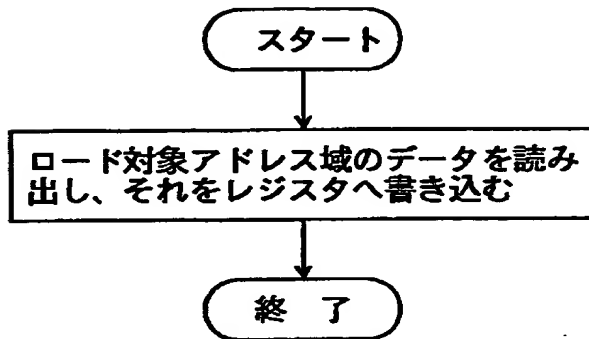
【図1】

汎用レジスタと浮動小数点レジスタとを有する従来の計算機の構成を示す図



【図 2】

ロード命令による動作を示すフローチャート



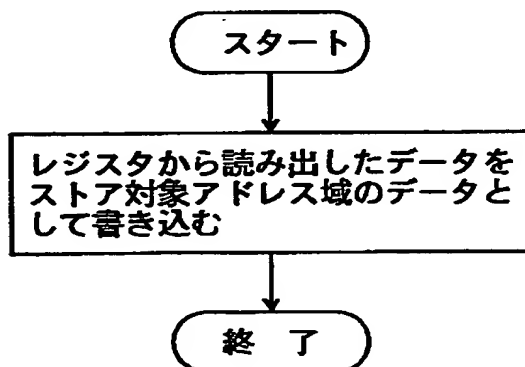
【図 3】

ロード命令の命令形式を示す図



【図 4】

ストア命令による動作を示すフローチャート



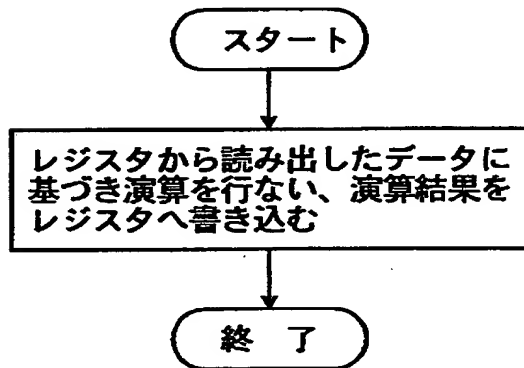
【図5】

ストア命令の命令形式を示す図



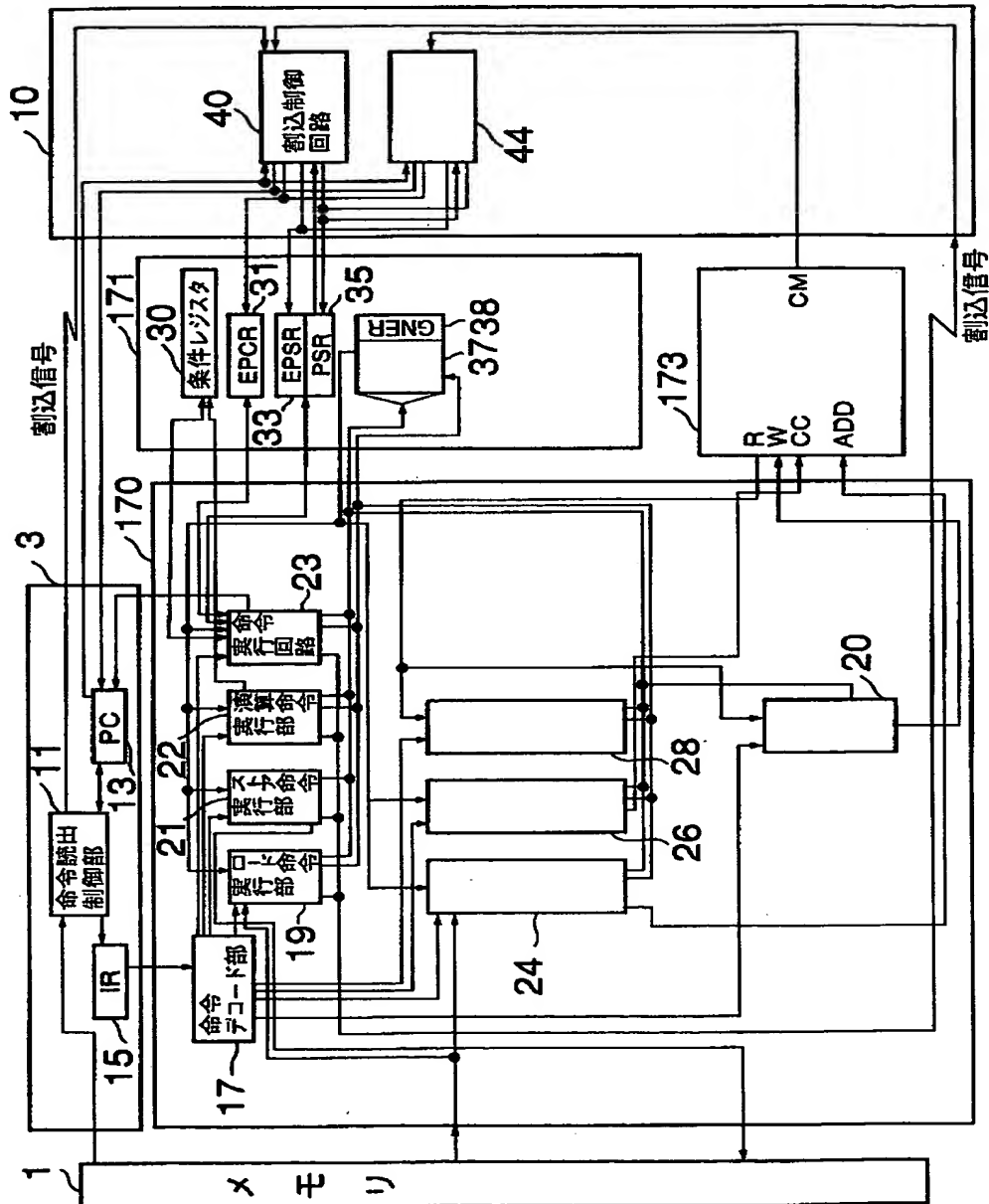
【図6】

演算命令による動作を示すフローチャート



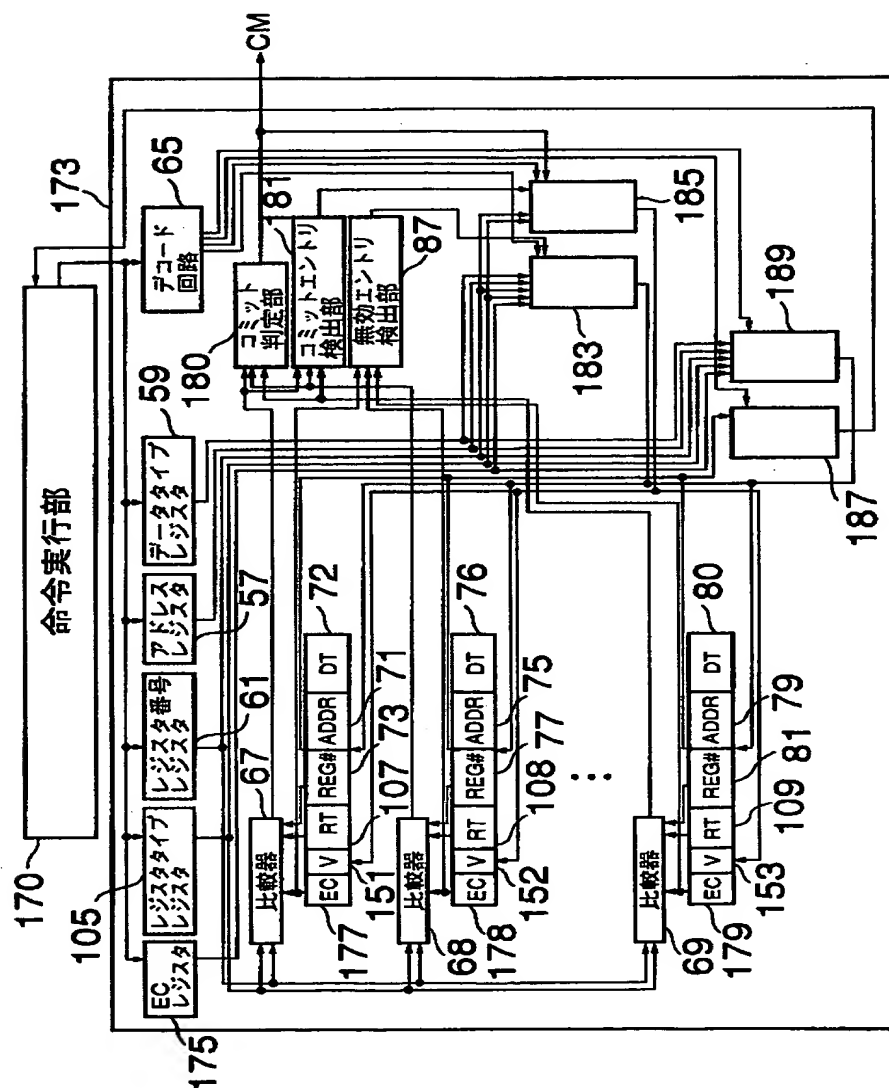
【図 7】

本発明の実施の形態1に係る計算機の構成を示す図



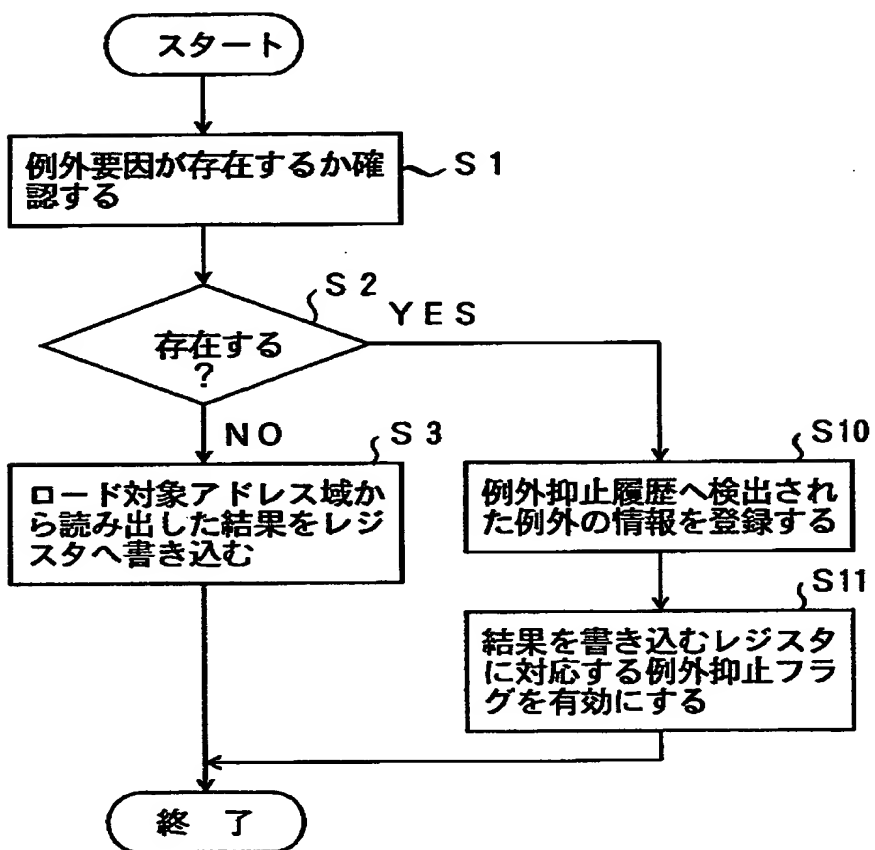
【图 8】

図7に示された履歴制御部の構成を示す図



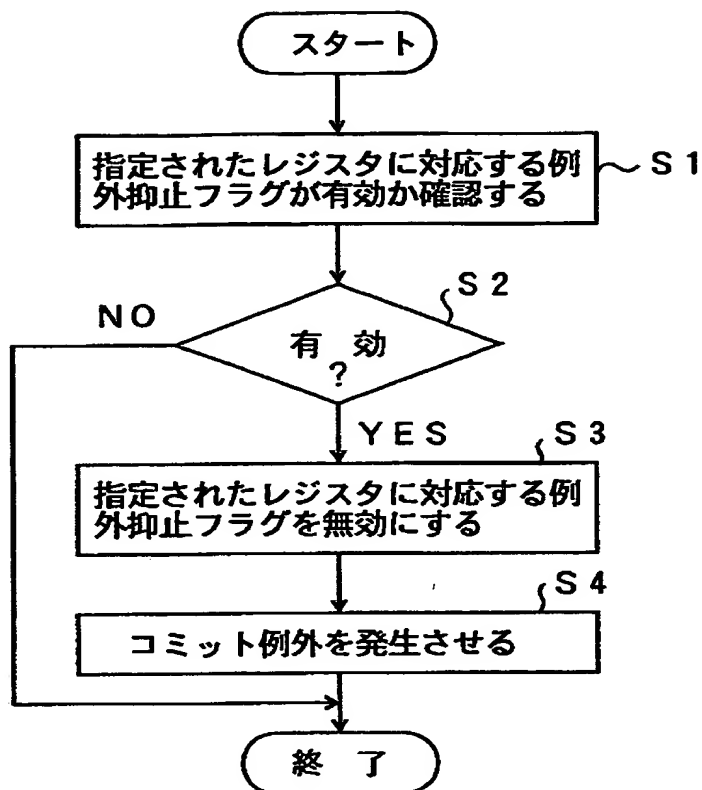
【図 9】

実施の形態 1 に係る例外抑止ロード命令
による動作を示すフローチャート



【図 1 0】

実施の形態 1 に係るコミット命令による動作を示すフローチャート



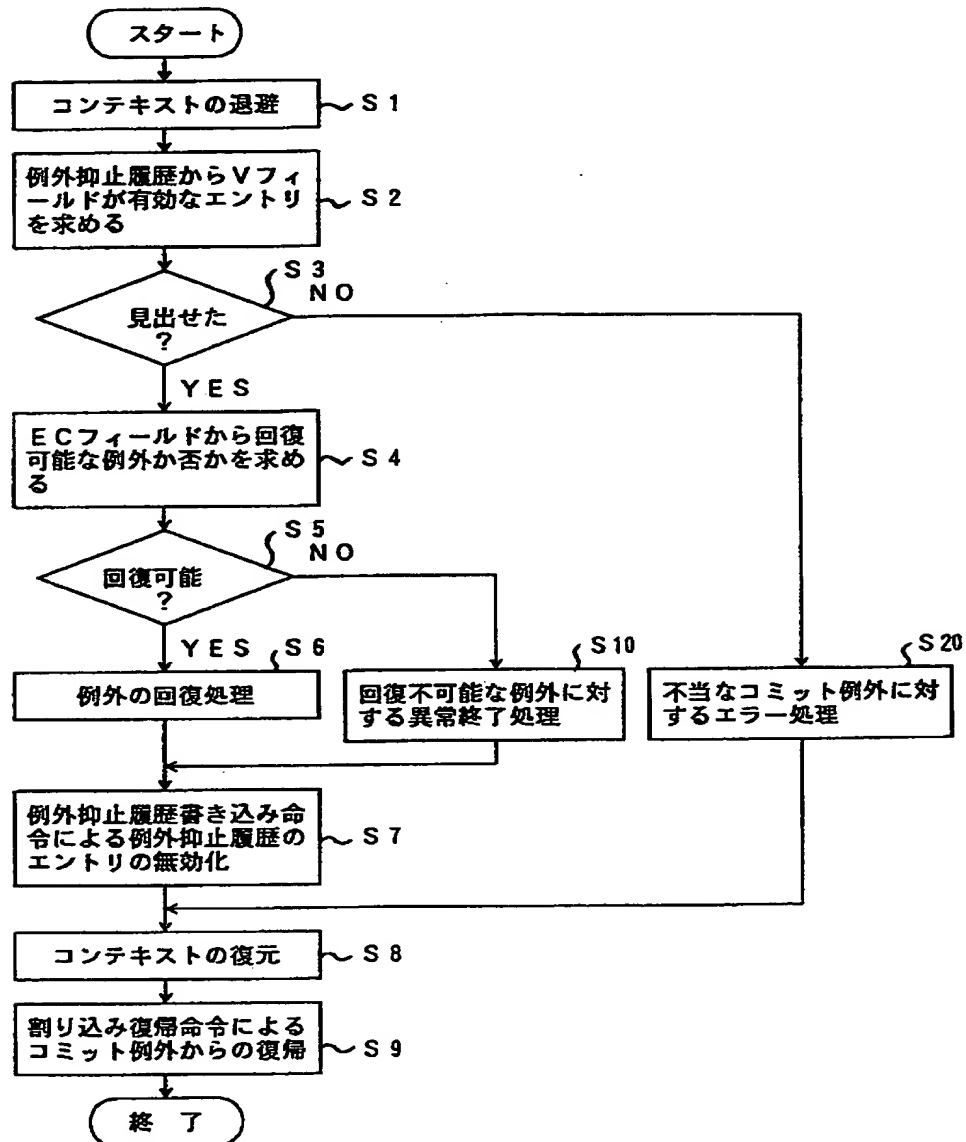
【図 1 1】

実施の形態 1 に係るコミット命令の命令形式を示す図



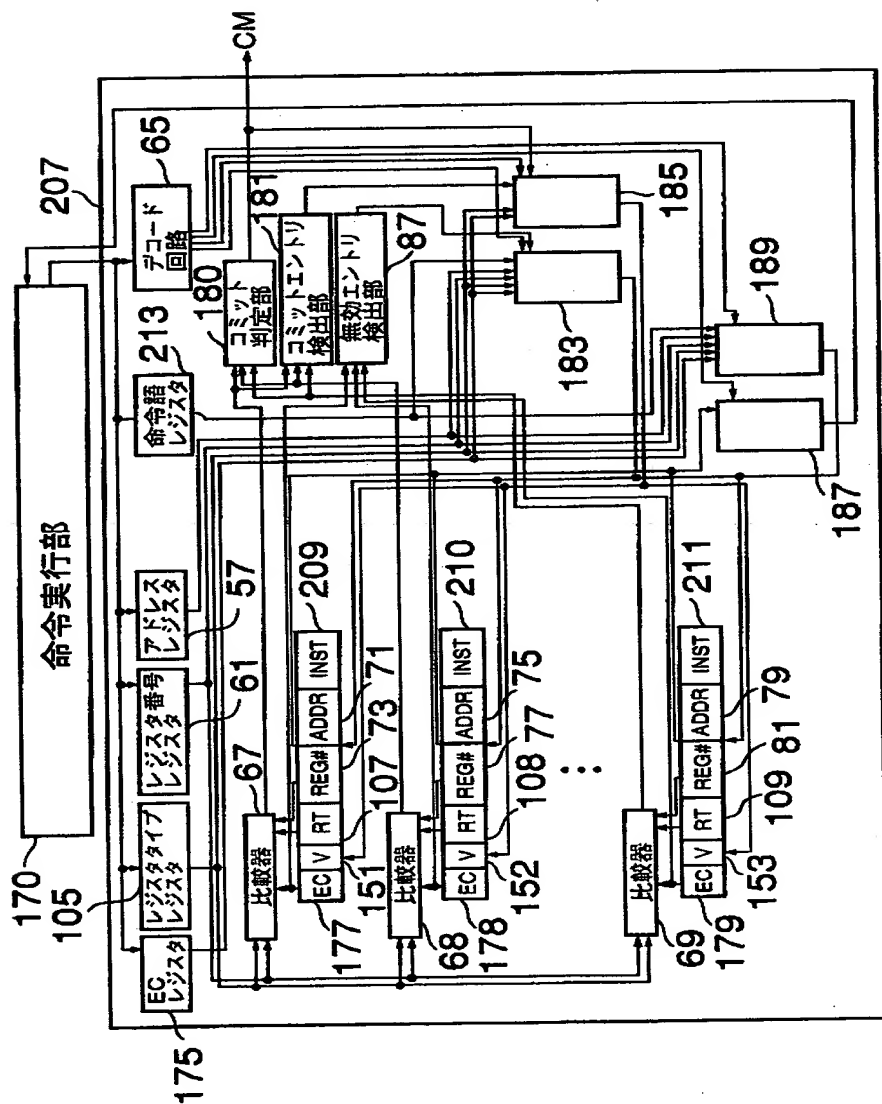
【図 1 2】

実施の形態 1 に係るコミット例外における
割り込み処理を示すフローチャート



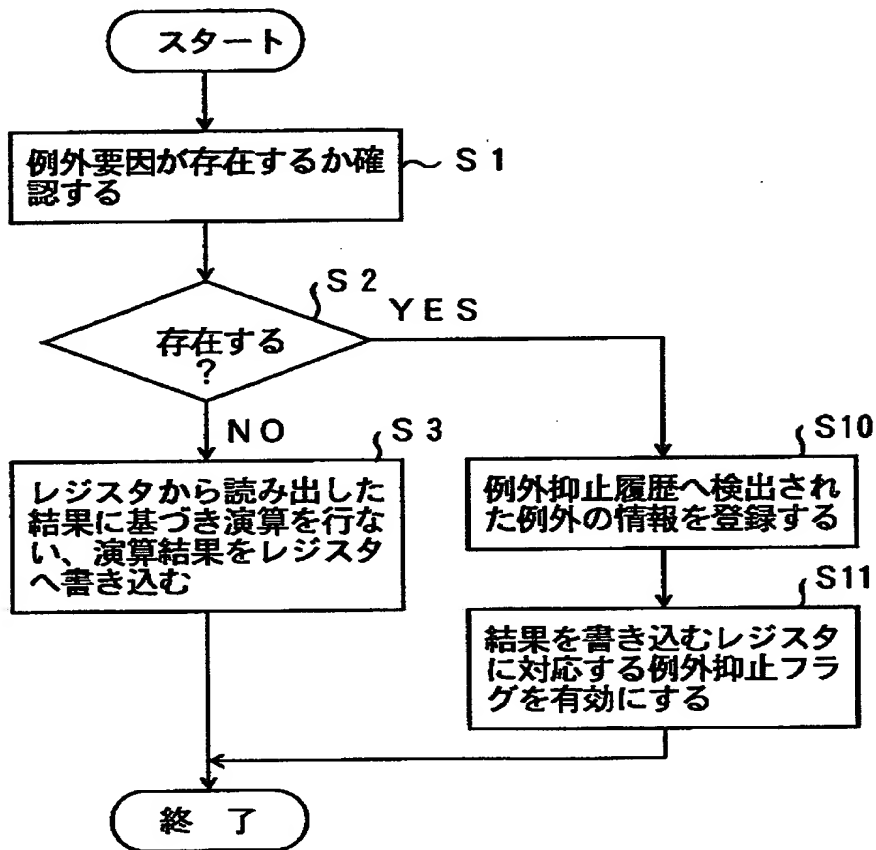
【图 14】

図13に示された履歴制御部の構成を示す図



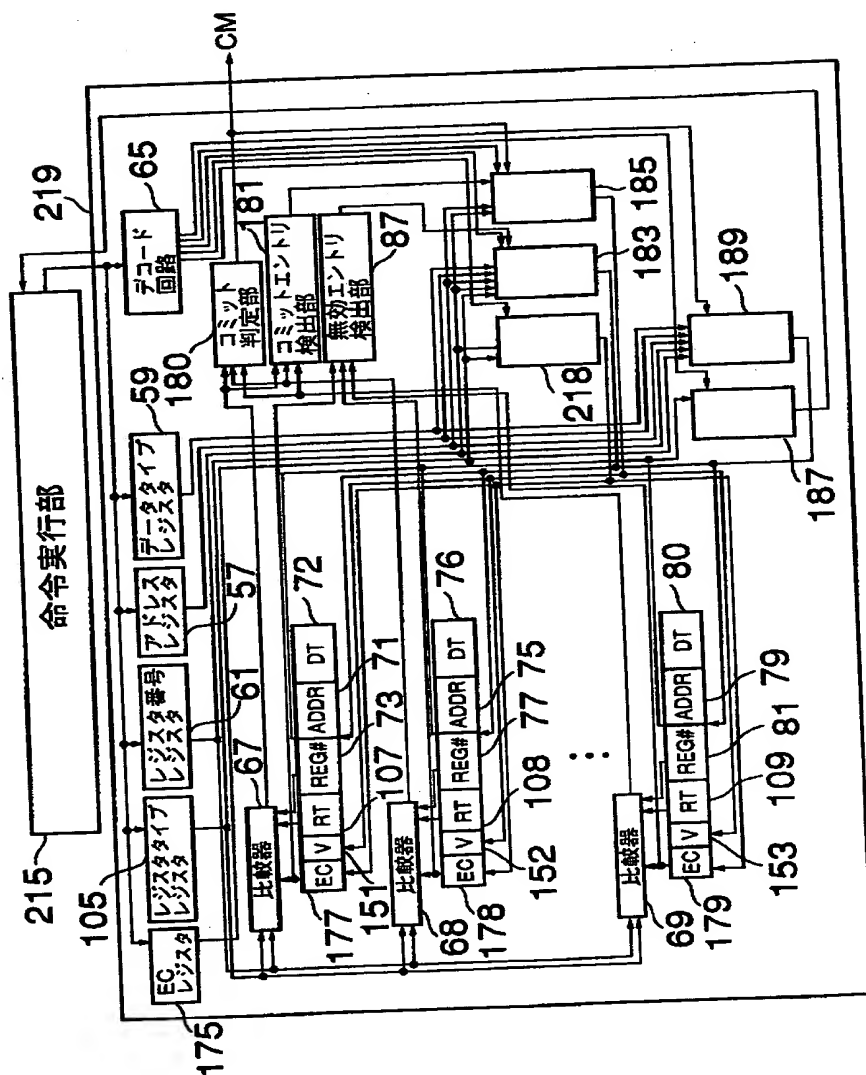
【図 1 5】

実施の形態 2 に係る例外抑止浮動小数点演算命令
による動作を示すフローチャート



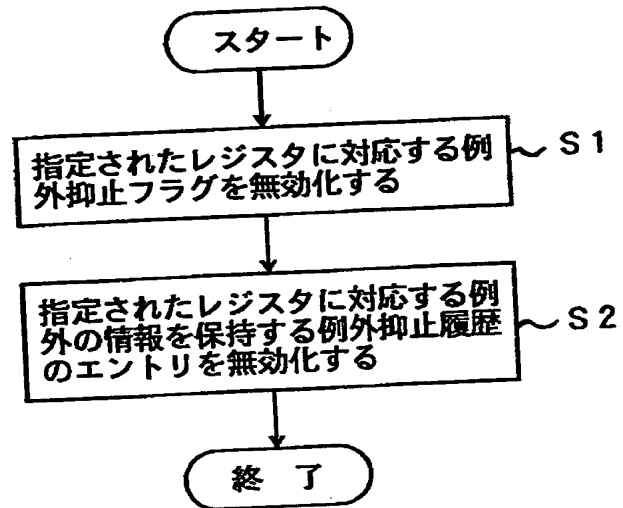
【図17】

図16に示された履歴制御部の構成を示す図



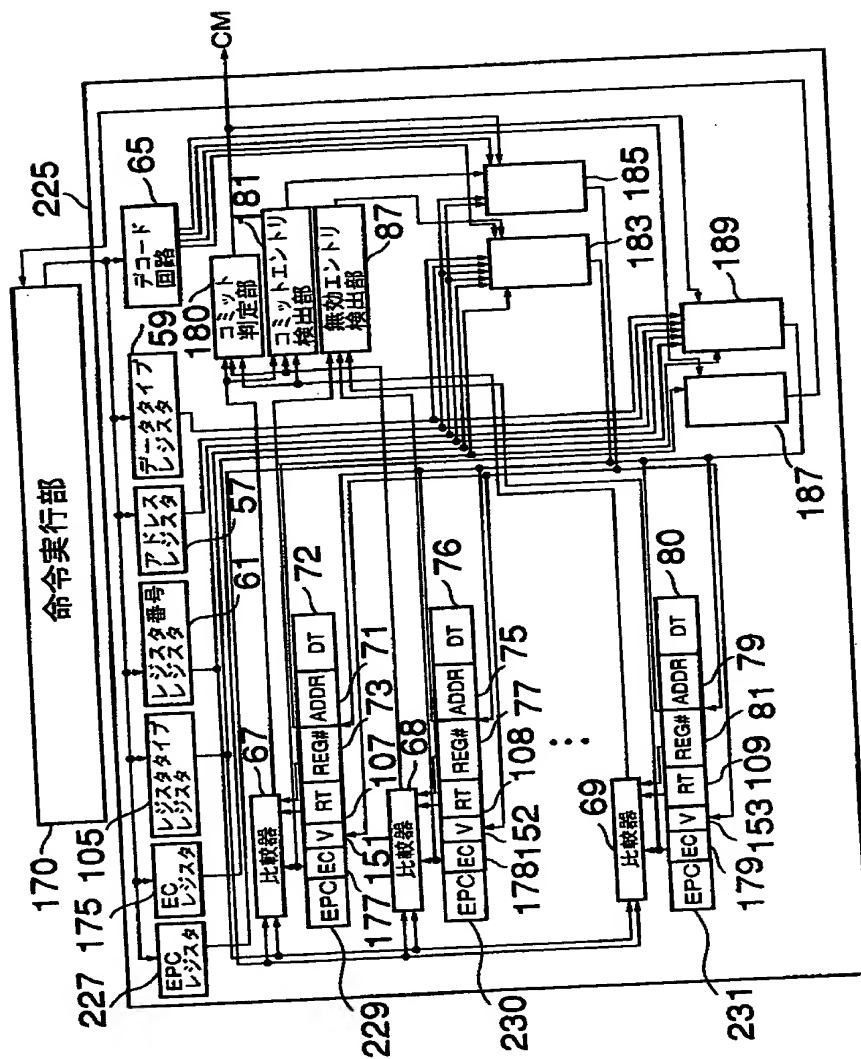
【図 1 8】

実施の形態 3 に係る例外抑止フラグ無効化命令による動作を示す図



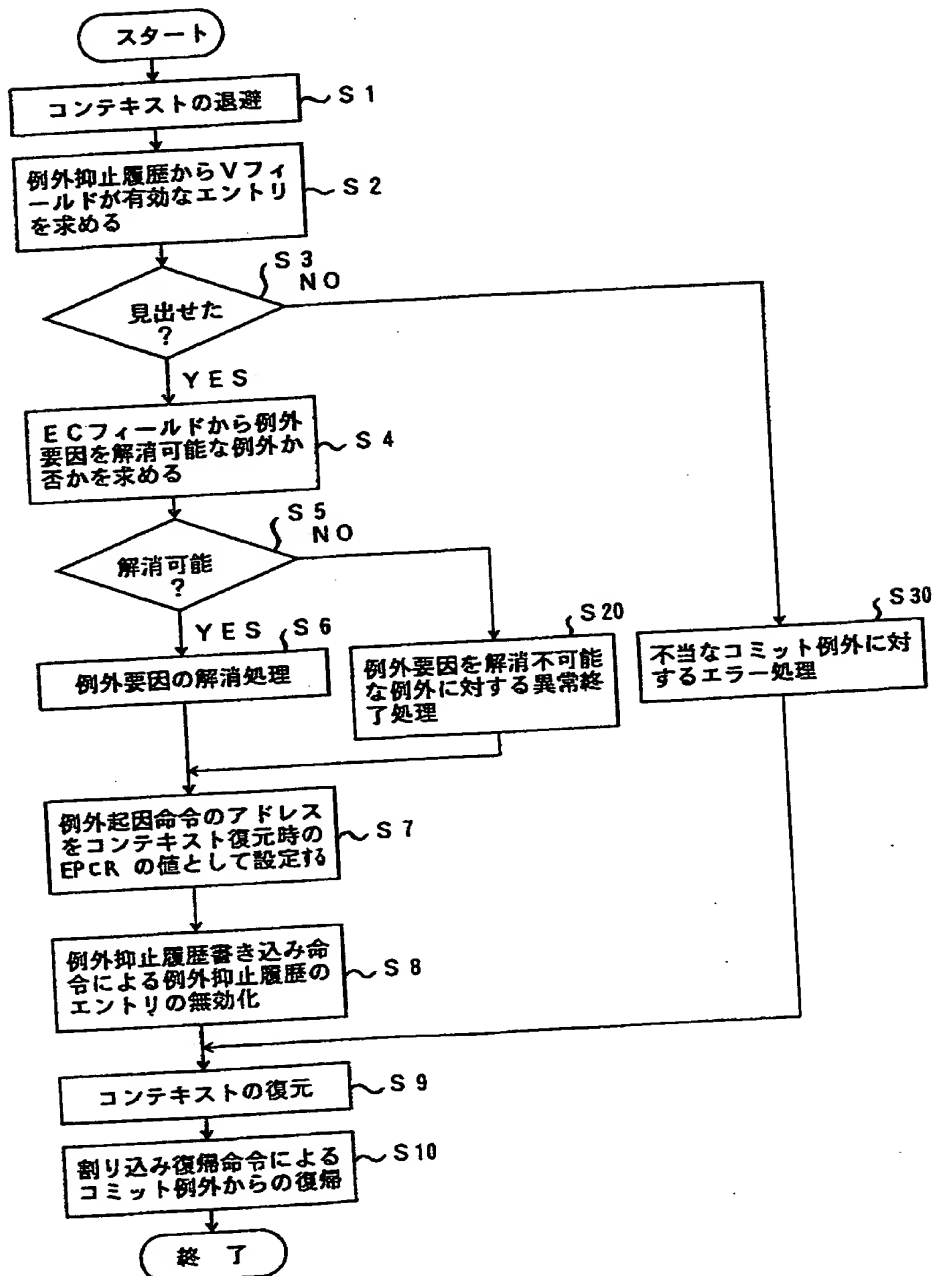
【図20】

本発明の実施の形態5に係る履歴制御部の構成を示す図



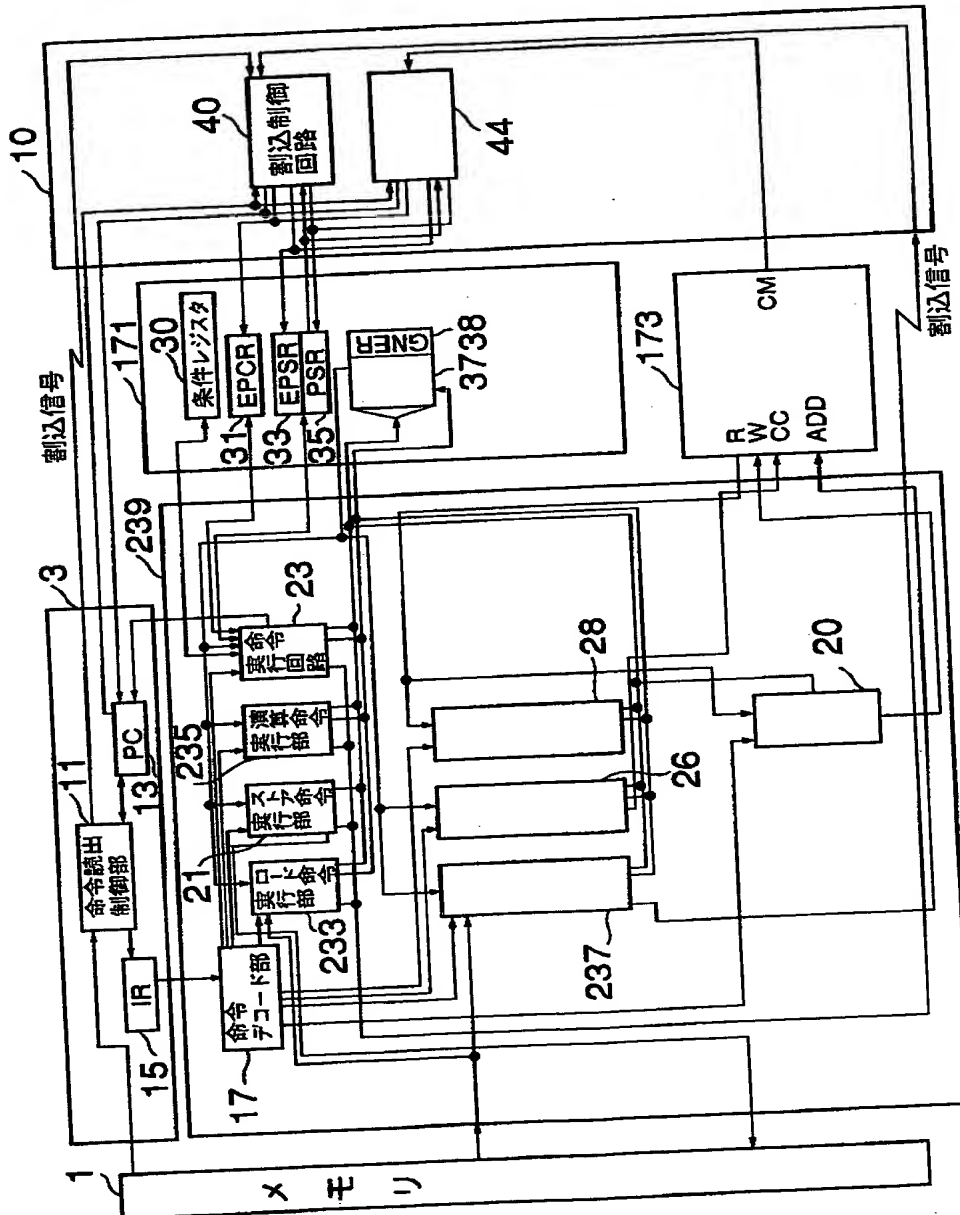
【図 21】

実施の形態 5 に係るコミット例外における
割り込み処理を示すフローチャート



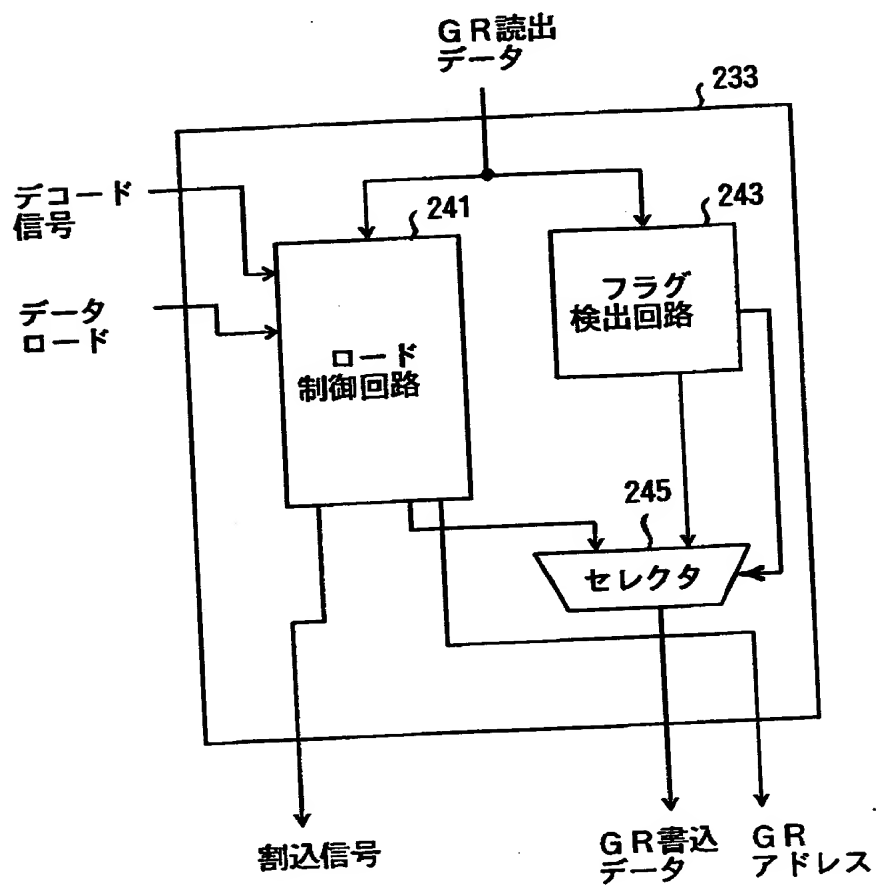
【図 22】

本発明の実施の形態6に係る計算機の構成を示す図



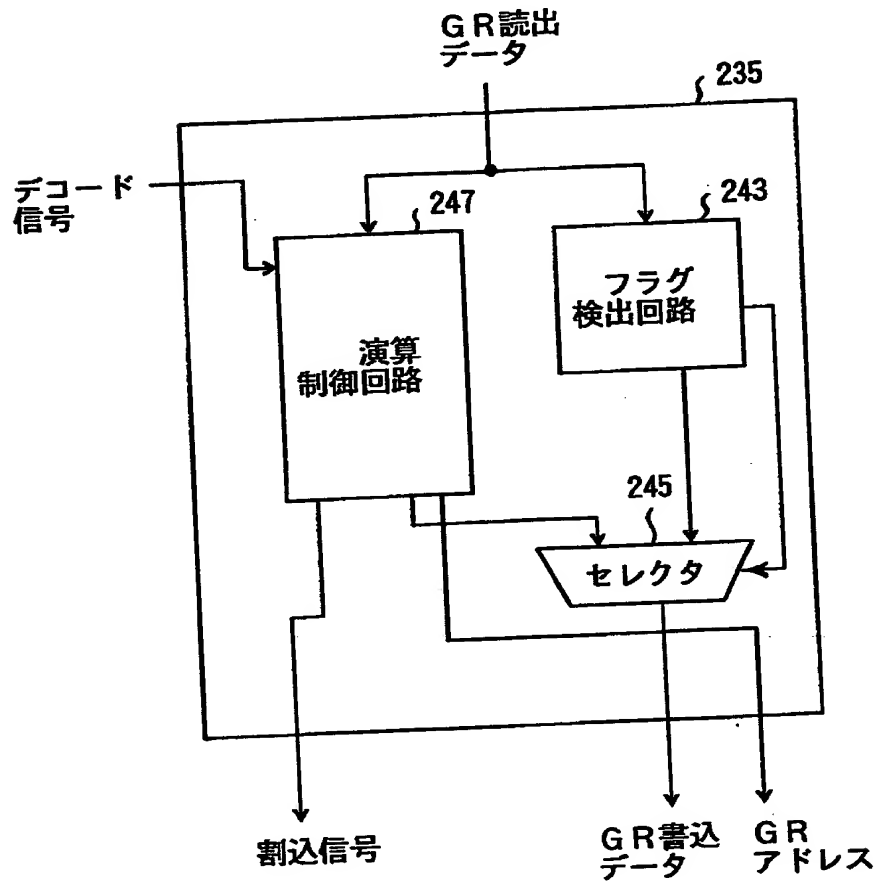
【図 23】

図 22 に示されたロード命令実行部の構成を示す図



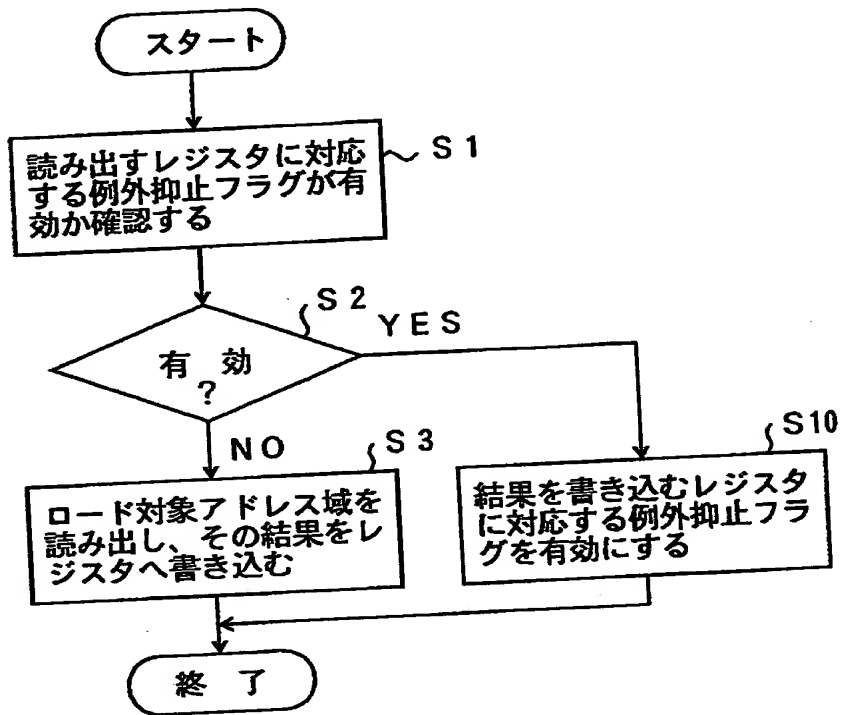
【図 24】

図 22 に示された演算命令実行部の構成を示す図



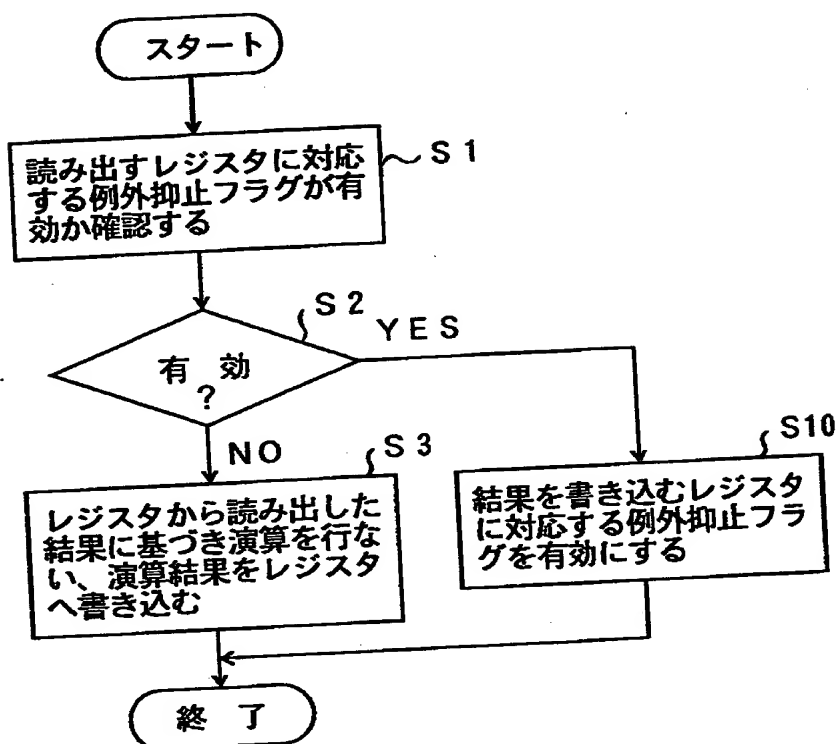
【図 25】

実施の形態 6 に係るロード命令による動作を示すフローチャート



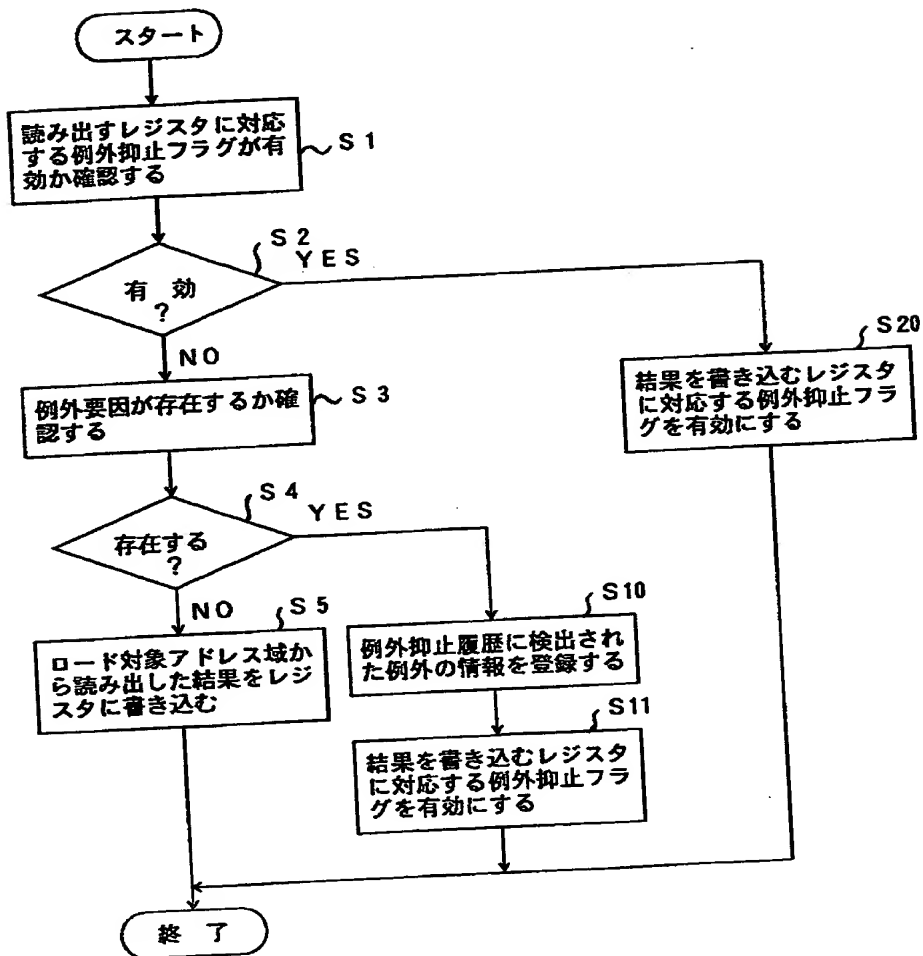
【図 2 6】

実施の形態 6 に係る演算命令による動作を示すフローチャート



【図 27】

実施の形態 6 に係る例外抑止ロード命令
による動作を示すフローチャート



【書類名】 要約書

【要約】

【課題】 動作の高速化により処理能力が高められた計算機とその制御方法を提供する。

【解決手段】 プログラムされた命令を実行する計算機であって、プログラムにおいて分岐命令より後置された命令を、分岐命令より先に実行するよう制御する命令読み出し制御部 1 1 と、該後置された命令の実行中に例外動作の必要性が検出された場合には、例外動作を留保する例外抑止ロード命令実行部 2 4 と、分岐命令を実行することにより選択された分岐先の命令の実行において、例外抑止ロード命令実行部 2 4 により留保された例外動作が必要とされる場合には、該例外動作を遂行するコミット例外割り込み制御部 4 4 と、例外動作が終了したときには、上記プログラムに復帰して例外動作の遂行が必要とされた命令の次の命令から順次実行するための履歴制御部 1 7 3 とを備えたことを特徴とする計算機を提供する。

【選択図】 図 7

特2000-043441

出 願 人 履 歴 情 報

識別番号

[000005223]

1. 変更年月日

1996年 3月26日

[変更理由]

住所変更

住 所

神奈川県川崎市中原区上小田中4丁目1番1号

氏 名

富士通株式会社